i-mobile for SP Application 広告表示SDK Version.2 Android版 設定資料



2015年3月版



Androidアプリに、広告を表示するためのSDKの設定方法を記載した技術資料です。

・対象となるサービス



- ・対象となる読者
 - ・アプリケーション開発者
- •広告表示方式
 - ・インタースティシャル広告
 - ・ウォール広告
 - ・バナー(バナー、ビックバナー、ミディアムレクタングル)広告
 - ・アイコン広告
 - ・テキストポップアップ広告

広告SDK利用方法

本書は、広告SDK を利用するために必要な以下の設定方法について説明します。

■ SDK について

- ・ SDK の説明
- ・スペック
- 管理画面の設定
 - ・ アプリケーションの登録
 - アプリケーションのカテゴリ登録
 - ・ 広告スポットの登録
 - ・フィルタの登録(任意)
 - ・ アプリケーション設定値の取得

■ アプリケーションの設定

- ・ SDK の適用
- ・マニフェスト設定
- サンプルアプリケーションの使用方法

■ 組み込み方法

《全画面広告》

- ・ 任意のイベントで広告を表示
- ・ 戻るボタン押下時に広告を表示
- ・ 起動時に広告を表示 (テキストポップアップ広告のみ)

《インライン広告》

- ・ レイアウトファイル (XML) に対するタグによる組み込み
- ・コードによる組み込み
- ・ デバイスの横幅に合わせて拡大表示する ※アイコン広告除く
- ・ アイコン広告表示形式のカスタマイズ
- AdMob Mediation への組み込み

《全広告共通》

- ・テストID
- ・テストモード
- ImobileSdkAdListener リスナーによる各種通知設定
- ・ バックグラウンド時に広告の取得を停止する方法

■ よくある質問と回答

- ・ 広告が表示されません
- ・ インライン広告を一時的に非表示にしたい
- ・ 初期の段階で広告が表示されない
- ・ Google Play Services の適用方法
- Android 4.0.x、4.1.xにおいて、アイコン広告の背景が透過されない
- SharedPreferencesに保存している情報について
- ・ SDKが取得している情報および送信している情報について

1. SDKについて

- SDK について
 - ・ SDK の説明

i-mobile SDK for Android を使用することにより、Android アプリケーション内から以下の広告を表示できるようになります。

- 全画面広告 インタースティシャル広告/ウォール広告/テキストポップアップ広告
- バナー広告 スマートフォンバナー/スマートフォンビッグバナー/ミディアムレクタングル/アイコン
- ・ スペック
 - SDK Version:2.0.7

■ 動作環境: Android version 2.3以上(APIレベル10) ※本バージョンよりandroid version 2.2以下は非対応となりました 本バージョンより、SDKを動作させるために、Google Play Services の適用が必要になりました ※未適用の場合は、本書の「よくある質問と回答 Google Play Services の適用方法」を参照いただき、適用をお願いいたします。



管理画面の設定 アプリケーションの登録



- ※1 アプリケーションがi-mobileに承認されるまでは実際の配信は行われません。
- ※2 マーケットURL は、アプリを特定するためのURL を入力してください。自身のアプリをマーケットにて検索した結果のURLで結構です。 まだマーケットに登録していない場合は、会社のURL等を入力してください。

管理画面の設定 アプリケーションのカテゴリ登録



管理画面の設定 広告スポットの登録(インタースティシャル)



管理画面の設定 広告スポットの登録(ウォール)

- 広告スポットの設定
 - (1) サイト/アプリ管理タブをクリック
 - 登録済みのアプリケーションのスポットリンクをクリック (2)
 - ③ 新規スポット登録ボタンをクリック
 - ④ スポット登録

広告スポット(アプリケーション内に配信する枠)の登録を行います。 広告スポット名を入力し、広告スポットサイズとしてウォールを選択後、 新規登録ボタンをクリックしてください。

レポート ① サイト/アプリ管理 フィルタ	収益の請求履歴 アカウン	・ト SDK ダウンロード	RTB とは
詳細 カテゴリ設定 広告スポット管理	パートナーサイトフィルタ	自社広告管理	
広告スポット登録 ・ <u>サイト/アプリ管理</u> ・ <u>テスト</u> 広告スポット名・?: な告スポットサイズ(単位:ピクセル)・: 新規登録	<u>アプリ</u> > <u>広告スポット管</u> フォール(全画面)	<u> 理</u> ▶ <u>広告スポット登</u> ・	<u>禄</u>



サイト/アプリ管理

検索

□ 名称

管理画面の設定 広告スポットの登録(テキストポップアップ)

	サイト/アプリ管理
■ 広告スポットの設定	 > セイトンフンDE位 ブラットフォーム: 全て 」 全て 」 ※
① サイト/アプリ管理タブをクリック	
② 登録済みのアプリケーションのスポットリンクをクリック	「 デストアプリーズ Test Ave
③ 新規スポット登録ボタンをクリック	(2) 編集
 4) スポット登録 	カテゴリ広告スポットフィルタ自社広告管理
広告スポット(アプリケーション内に配信する枠)の登録を 広告スポット名を入力し、広告スポットサイズとしてテキス 広告の表示頻度の設定を行い、新規登録ボタンをクリッ	:行います。 ペトポップを選択後、 クしてください。
レポート サイト/アプリ管理 フィルタ 収益の請求履歴 アカウン	ト SDK ダウンロード RTB とは
詳細 カテゴリ設定 広告スポット管理 パートナーサイトフィルタ	自社広告管理
広告スポット登録 ・ サイト/アプリ管理 ・ テストアプリ ・ 広告スポット管理 広告スポット名 * ?: ・ 広告スポット名 * ?: ・ 広告スポットサイズ(単位:ピクセル) *: ・ 表示頻度(スキップ回数)* ?: ・ 新規登録 ・	里 ♪ 広告スポット登録 ▼ する頻度として「スキップ回数」の設定が可能です。 数を設定した場合は、設定された回数分、広告表示処理をスキップします。

管理画面の設定 広告スポットの登録(バナー、アイコン)



管理画面の設定 フィルタの登録(任意)



管理画面の設定 アプリケーション設定値の取得

- アプリケーション設定値の取得
 - ① サイト/アプリ管理タブをクリック
 - ② スポット管理タブをクリック
 - ③ スポット一覧にてアプリ設定取得ボタンをクリックしてください。
 - ④アプリケーション設定値の取得
 - 画面上の項目をメモしておいてください。この後に説明するアプリケーションへの組み込み時に必要となります。

	(1)						-
	レポート サイト/アプリ管理	フィルタ 収益の請求層歴 アカウント SDK ダウ	フンロード				

	詳細 カテコリ設定 広告。	ペパット管理 パートナーリイトフィルタ 自社広告管理					
	広告スポット管理 サイト/アプリ管理	 ・ テストアプリ ・ 広告スポット管理 					
	停止 再開 削除 ※ 「削除」をされますとデータを復元できま	きせんので、ご注意ください。					新規広告スポット
	□ 広告スポット	広告スポットサイズ(単位:ピクセル)	広告表示方式 3	広告タグ	アニメーションタイプ	リフレッシュレート	表示
	□ <u>インタースティシャル</u>	インターステーシャル(300x250 / 600x500)	インライン	アプリ設定取得	None	リフレッシュなし	レポート
	1.ご自身のアプリに貼りたい広告の	サイズごとに右上にある ^{「新規広告スポット} のボタンから広告スポ	ットを追加します。				
	2.次にアプリ設定取得のボタンから	設定を取得し、それをご自身のアプリに導入してください。					
	3.きちんと広告が表示されたら設定	は成功です。					
	レポート サイト/アプリ管理 こ	フィルタ 収益の請求履歴 アカウント SDK ダウンロード					
	詳細 カテゴリ設定 広告スポ	ット管理 パートナーサイトフィルタ 自社広告管理					
4	 アプリ設定取得 サイト/アプリ管理 	- <u>テストアプリ</u> ・ 広告スポット管理 ・ インタースティ	<u>シャル > アプリ:</u>	受定取得			
Ψ	アプリ設定値						
	パブリッシャーID: ****						
	メディアID: ****						
	スポットID: *****						
	SDKダウンロード						
	SDKのダウンロードはこちらです						
	<u>55.000000 - Hacoboo</u>						

アプリケーションの設定 SDKの適用

■SDKの適用

ダウンロードしたSDKを、Eclipseのプロジェクトにて参照できるようにします。 この作業によって、i-mobile SDK for Androidが利用可能となります。

① libsフォルダの作成

Eclipseのプロジェクト内にSDKを格納するlibsフォルダを作成してください。

② i-mobileSDK.jarの配置
 ①で作成したlibsフォルダに、取得した「imobileSdkAds.jar」を配置してください。

③ ビルドパスの設定 <u>**ADT revision 17以降を使用している場合は不要です</u>
 配置したSDKをプロジェクトのビルドパスに含めてください。
 プロジェクトを右クリックし、プロパティを選択後、jar追加をクリックし先ほど
 配置したSDKを選択してください。

🙆 AndroidSample のプロパティ	(-		
フィルター入力	Java のビルド・パス		⇔ • ⇔ • •
 > リソース Android Android AndyEdt ツール Checkstyle DBViewer ブラグイン FindBugs Java エディター Java コンパイラー Java コンパイラー Java のビルド・パス = Javadoc ロケーショ: Limy Tomcat WikiText サーバー タスク・タガ タスク・リポジトリー ビルダー プロジェクト・ファt プロジェクト参照 	● ソース(S) (ご ブロジェクト(P) ▲ ライブラリー(L) 🌭 順序およびエクスポート(O) ビルド・パス上の JAR およびクラス・フォルダー(T): ▷ 🚵 Android 1.6	3	Jar 追加(J) 外部 Jar 追加(X) 変数の追加(V) ライブラリーの追加(A) クラス・フォルダーの追加(C) 外部クラス・フォルダーの追加(D) 端集(E) 時去(R) JAR ファイルのマイグレーション(M)
4 III +	•		•
?		Oł	キャンセル



proguard.cfg

アプリケーションの設定 マニフェスト設定

■ マニフェストファイル編集

プロジェクト直下の「AndroidManifest.xml」を編集します。

※既に設定済みの場合は、この対応は不要です

① Google Play Services の設定

事前にGoogle Play Services の適用が必要になります。

※未適用の場合は、本書の「よくある質問と回答 Google Play Services の適用方法」を参照いただき、適用をお願いいたします。

② パーミッションの設定

以下のパーミッションを付与してください。

INTERNET : ネットワーク経由にて広告を取得するためのパーミッションです。 ACCESS_NETWORK_STATE:ネットワークの状態を確認するためのパーミッションです。

【マニフェストのサンプル】



サンプルアプリケーションの使用方法

サンプルアプリケーションプロジェクトをEclipse等で読み込み実行してください。 設定ダイアログから、各種IDを設定することにより、設定されたID で広告の表示を行います。 未設定の場合は、デフォルト値としてテストID を使用して広告の表示を行います。



🚭 👍 🛶 🌵 🖙 🐫 🍘 🖧 🕴 🙃 🖬 79% 📓 17:34

 (320×50)

《全画面広告》 組み込み方法 【任意のイベントで広告を表示する】

このサンプルでは、ボタンが押されたときに広告を表示します

import jp.co.imobile.sdkads.android.ImobileSdkAd; <		· 1
<pre>@Override protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity_detail);</pre>		
//1. 広告スポットの登録 ImobileSdkAd. <i>registerSpotFullScreen</i> (MainActivity.this, PUBLISHER_ID, MEDIA_ID, SPOT_ID); ◀ //2. 広告取得開始 ImobileSdkAd. <i>start</i> (SPOT_ID);◀	(2 3
Button btnShowAd = (Button)findViewById(R.id.btnShowAd); btnShowAd.setOnClickListener(new View.OnClickListener(){ @Override public void onClick(View v) {	(4
@Override protected void onDestroy() { //Activity廃棄時の後処理 ImobileSdkAd.activityDestory; ◀	(5
}		

① ImobileSdkAd をインポートします。

- ② ImobileSdkAd にスポットを登録します。(PUBLISHER_ID, MEDIA_ID, SPOT_IDには、アプリケーションの設定取得[2/2]で取得した値を設定してください)
- ③ ②で登録したスポットの広告取得を開始します。(SPOT_IDには、アプリケーションの設定取得[2/2]で取得した値を設定してください)
- ④ 広告を表示します。(SPOT_IDには、アプリケーションの設定取得[2/2]で取得した値を設定してください)
- 5 Activity 廃棄時に、SDK側の後処理を実行します。

※この処理を記述しない場合、「has leaked window com.android.internal.policy.impl.PhoneWindow\$DecorView」エラーが発生することがあります

⑤の記述は、広告を表示する全てのActivityに<u>必ず記述</u>してください。
 また、②および③については、必ずonCreateに記述し、⑤については、必ずonDestroyに記述してください。
 インタースティシャルとウォール広告で広告の表示方法に違いはありません。
 指定されたスポットIDがインタースティシャルスポットの場合は、インタースティシャル広告が表示され、ウォールスポットの場合は、ウォール広告が表示されます。

《全画面広告》 組み込み方法 【戻るボタン押下時に広告を表示する】

このサンプルでは、戻るボタン(Backキー)が押されたときに広告を表示します



《全画面広告》 組み込み方法 【起動時に広告を表示する(テキストポップアップ広告のみ)】

テキストポップアップ広告は、起動時に広告の表示処理を入れることが可能です。

※ インタースティシャル広告やウォール広告についても同様の実装は可能ですが、テキストポップアップ広告と比べ、広告の取得に時間がかかるため、 多くの場合、「showAd」呼び出し時に広告表示準備が完了していないため、広告の表示がされません。

このサンプルでは、起動(Activity 作成)時に広告を表示します

import jp.co.imobile.sdkads.android.ImobileSdkAd; import jp.co.imobile.sdkads.android.ImobileSdkAdListener;	
<pre>@Override protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity_detail);</pre>	
//1. 広告スポットの登録 ImobileSdkAd. <i>registerSpotFullScreen</i> (MainActivity.this, PUBLISHER_ID, MEDIA_ID, SPOT_ID); //2. 広告取得開始 ImobileSdkAd. <i>start</i> (SPOT_ID); //3. 広告表示 ImobileSdkAd.showAd (this,SPOT_ID);◀	2
@Override protected void onDestroy() {	

- ① ImobileSdkAdListener をインポートします。
- ②「2.広告の取得開始」に続けて、showAdを呼び出します。
 ※ テキストポップアップ広告は、「showAd」が呼び出されてから約2秒以内に広告の取得および表示ができない場合は、広告の表示をキャンセルします。

《インライン広告》 組み込み方法 【レイアウトファイル(XML)に対するタグによる組み込み】

このサンプルでは、画面の最下部中央に広告を表示します

<framelayout <br="" xmlns:android="http://schemas.android.com/apk/res/android">xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent" android:layout_height="match_parent" tools:context=".MainActivity" ></framelayout>	
<jp.co.imobile.sdkads.android.imobileinlineview android:layout_width="wrap_content" android:layout_height="wrap_content" android:layout_gravity="bottom center_horizontal"</jp.co.imobile.sdkads.android.imobileinlineview 	 1
	\otimes
sid="SPOT_ID" />	 3 4

- ① タグとして「jp.co.imobile.sdkads.android.lmobileInlineView」を指定します
- ② PUBLISHER_IDにパブリッシャーIDを設定します。(パブリッシャーIDは、アプリケーションの設定取得で取得した値を設定してください)
- ③ MEDIA_IDにメディアIDを設定します。(メディアIDは、アプリケーションの設定取得で取得した値を設定してください)
- ④ SPOT_IDにスポットIDを設定します。(スポットIDは、アプリケーションの設定取得で取得した値を設定してください)

ImobileInlineViewはRelativeLayoutを継承しているため、RelativeLayoutと同様に扱うことが可能です(レイアウトに関して制約を課すものはありません)。

旧SDKとは、クラス名および、広告表示処理の記述方法が変わっておりますので、旧SDKからの移行の場合は記述の変更が必要です。

《インライン広告》 組み込み方法 【コードによる組み込み】

このサンプルでは、画面表示時に広告を表示します

import jp.co.imobile.sdkads.android.ImobileSdkAd; <		1		
<pre>@Override protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity_detail);</pre>				
// 【表示する座標を指定する場合】 // 1.広告スポットの登録 ImobileSdkAd.registerSpotInline(this, PUBLISHER_ID, MEDIA_ID, SPOT_ID); ◀ // 2.亡告取得問始		2		
// と)及日本(所) ImobileSdkAd.start(SPOT_ID); ●		3		
// 3.広告表示(表示する座標を指定) ImobileSdkAd.showAd(this, SPOT_ID, X座標, Y座標, true); ◀·······		4		
// 【表示するViewGroupを指定する場合】 // 1.広告スポットの登録				
ImobileSdkAd.registerSpotInline(this, PUBLISHER_ID, MEDIA_ID, SPOT_ID); ◀		2		
ImobileSdkAd.start(SPOT_ID);		3		
// 3.広告表示(表示するViewGroupを指定) ViewGroup adLavout = (ViewGroup)findViewBvld(R.id.adLavout): ◀		(5)		
ImobileSdkAd.showAd(this, SPOT_ID, adLayout); @Override android:layout_width="wrap_content" protected void onDestroy() { android:layout_height="wrap_content" // Activity廃棄時の後処理 android:layout_gravity="bottom center_horizontal" />				
ImobileSdkAd.activityDestory(); super.onDestroy(); }		6		
旧SDKとは、クラス名および、広告表示処理の記述方法が変わっておりますので、旧SDKからの移行の場合は記述の変更が必要です。 また、表示するViewGroupを指定する方法を使うことにより、従来のSDKの様に広告をLayoutとして取り扱うことが可能です。				
① ImobileSdkAd をインポートします。 ② ImobileSdkAd にスポットを登録します。(PUBLISHER_ID, MEDIA_ID, SPOT_IDには、アプリケーションの設定取得で取得した値を設定してください)				
※バナーを表示するスポットの場合は、ImobileSdkAd.registerSpotInlineを使用してください。 ③ ②で登録したスポットの広告取得を開始します、(SPOT IDには、アプリケーションの設定取得で取得した値を設定してください)				
④ 「「「「「」」」」」」」」」」」」」」」」」」」」」」」」」」)」)」)」)」				
※第5引数にtrueを設定した場合は、指定した座標に対してdpi変換を行い、falseを設定した場合はdpi変換を行いません。				
5) 指定したViewGroupに広告を表示します。(SPOT_IDには、アフリケーションの設定取得で取得した値を設定してください)				

(6) ACTIVITY廃業時に、SDK側の後処理を実行します。
 ※この処理を記述しない場合、「has leaked window com.android.internal.policy.impl.PhoneWindow\$DecorView」エラーが発生することがあります。

- ①~⑥の記述は、広告を表示する全てのActivityに必ず記述してください。
- また、②および③については、必ずonCreateに記述し、⑥については、必ずonDestroyに記述してください。

スマートフォンバナー、スマートフォンビッグバナー、ミディアムレクタングル、で広告の表示方法に違いはありません。指定されたスポットIDのサイズに対応したバナー広告が表示されます。

《インライン広告》 組み込み方法 【デバイスの横幅に合わせて拡大表示する※アイコン広告除く】

バナー広告(320x50)、ビックバナー広告(320x100)は、デバイスの幅が320(mdpiでは本来360)のデバイスに合わせた広告サイズですが、 最近では、xxxhdpi(1080px1920px)のデバイスも増えてきており、従来のバナー広告(320x50)、ビックバナー広告(320x100)を表示した場合、 大きく余白ができてしまいます。



showAdメソッドの第6引数に「true」を指定した場合、バナーを横幅いっぱいまで拡大表示し、余白ができない様に表示することができます。 ※拡大されるサイズは、デバイスサイズの短い方のサイズまでです。



拡大した場合、縦横比を維持するために、横幅に合わせて縦幅も併せて拡大されるため、必要に応じてY座標の調整(上方向)を行っています。



※Y座標の調整は、座標指定でshowAdを実行した場合のみおこなわれ、ViewGroupを指定した場合は行われません。

Proprietary and Confidential to i-mobile Co., Ltd. All Rights Reserved

のデバイス

《インライン広告》 アイコン広告表示形式のカスタマイズ

アイコン広告の場合、前述の組み込み方法で組み込みを行った場合、以下の様に表示されます。



この表示形式は、ImobileIconParamsにパラメータを設定し、showAdメソッドに渡すことにより変更することができます。 変更可能なパラメータは、以下の通りです。

※サイズ、位置を指定するパラメータの数値は、全てdpi値になります。

パラメータ名	説明	記述サンプル(XML)	記述サンプル(コード)	デフォルト値
表示個数	アイコンの表示個数を1~6範囲で設定することができます。 アイコンを表示するためには、1つあたり75x75の領域が必要になります。 そのため、アイコンを3つ表示する場合は、Width:75x3 Height:75 の領域 が必要になります。 ※4以上を指定した場合は、デバイスの短い方を幅を領域(ポートレイト表示 の場合は横幅全て)として表示されます。 但し、アイコンサイズ、タイトル表示の有無、タイトルのフォントサイズ等のパ ラメータを設定した場合、必要な表示領域は設定値に伴って変化します。	iconNumber="1" 例1: iconNumber = 3, iconSize = 8 Width: (80 [iconSize] + 18 [デフォノ Height: 80 [iconSize] 例2: iconNumber = 4, iconSize = 6 Width: 320 [デバイスの短い方の領紙 Height: 60 [iconSize] + 4 [アイコンス	params.seticonNumber (1); 0, タイトルを非表示に設定した場合 いのマージン]) * 3 0, タイトルを表示, フォントサイズを8に設定した場合 或] ヒタイトルの間隔] + 8 [フォント] + 1 [影] + 3 [下部マージン]	4
表示領域サイズ	アイコンの表示領域の横幅のサイズを設定することができます。 任意の幅を指定したい場合に設定してください。 例:ランドスケーブのアプリで横幅全てを使用してアイコンを表示したい場合	iconViewLayoutWidth="75"	params.setIconViewLayoutWidth (75) ;	デバイスの 短い方のサイズ
アイコンサイズ	アイコンの画像サイズを設定することができます。 設定可能な最小値は47です。	iconSize ="57"	Params.setIconSize (57) ;	57 (領域に収ま らない場合47)
タイトル表示 有無	タイトルを表示/非表示を設定することができます。	iconTitleEnable="true"	params.setIconTitleEnable (true) ;	true
タイトル フォントサイズ	タイトルのフォントサイズを指定することができます。 設定可能な最小値は8です。	iconTitleFontSize="10"	params.setIconTitleFontSize (10);	10
タイトル フォントカラー	タイトルのフォントカラーをRGB形式で指定することができます。	iconTitleFontColor="#FFFFFF"	params.setIconTitleFontColor ("#FFFFFF");	#FFFFFF(白)
アイコン画像と タイトルの間隔	アイコン画像とタイトルの間隔を設定することができます。	iconTitleOffset="4"	params.setIconTitleOffset (4) ;	4
影付き 有無	タイトルに影を付けるか否かを設定することができます。	iconTitleShadowEnable="true"	Params.setIconTitleShadowEnable (true)	true
影の色	タイトルの影の色をRGB形式で指定することができます。	iconTitleShadowColor="#000000"	params.setIconTitleShadowColor ("#000000") ;	#000000(黒)
影の位置(X座標)	タイトルの影の位置(X座標)を指定することができます。	iconTitleShadowDx="1"	params.setIconTitleShadowDx (1);	1
影の位置(Y座標)	タイトルの影の位置(Y座標)を指定することができます。	iconTitleShadowDy="1"	params.setIconTitleShadowDy (1);	1

《インライン広告》アイコン広告表示形式のカスタマイズ

このサンプルでは、1つのアイコン広告をタイトル色を黒、タイトルの影無しで表示します。

【 XMLによる記述 】



【 コードによる記述 】

- **①** ImobileIconParams をインポートします。
- ② ImobileIconParams のインスタンスを生成します。
- ③ ②で生成したImobileIconParams のインスタンスのアイコンの表示個数を設定するパラメータに、「1」を設定します。
- ④ ②で生成したImobileIconParams のインスタンスのアイコンタイトルの色を設定するパラメータに、「#000000」(黒)を設定します。
- 5 ②で生成したImobileIconParams のインスタンスのアイコンタイトルの影付きの有無を設定するパラメータに、「false」を設定します。
- ⑥ showAdの引数として、②で生成したインスタンスを設定します。

旧SDKとは、クラス名および、広告表示処理の記述方法が変わっておりますので、旧SDKからの移行の場合は記述の変更が必要です。

《インライン広告》 AdMob Mediation への組み込み

同梱されているAdMob Mediation アダプタを利用することにより、AdMob Mediationに対して、 容易にi-mobileの広告を配信することができます。

AdMobメディエーションの設定については、以下のURL を参照してください。

https://support.google.com/admob/v2/answer/3063564?hl=ja&ref_topic=3063091&rd=1

AdMobメディエーションの組み込み方法については、以下のURLを参照してください。

https://developers.google.com/mobile-ads-sdk/docs/admob/mediation

■ AdMobMediation アダプタの適用

※以下は、「アプリケーションの設定 マニフェスト設定」まで完了していること、およびAdMob Mediationの設定が完了していることがが前提となります。 ダウンロードしたSDKを、Eclipseのプロジェクトにて参照できるようにします。 この作業によって、AdMobMediation アダプタが利用可能となります。

① i-mobileSDK.jarの配置

libsフォルダに、「i-mobileSDKMediationAdapter.jar」を配置してください。

② マニフェストファイルの記述の変更

下記のサンプルを参考に、マニフェストファイルに「com.google.android.gms.ads.AdActivity」を追加してください。 ※ AdMobの導入をしている場合は、既に記述があるはずですので、この対応は不要です

【マニフェストのサンプル】



また、旧来のSDK Version1.x 用のアダプタとの共存はできません。

《全広告共通》テストID

テストIDとは、スポット登録前やスポット登録直後は広告の配信がされないため、その場合に使用するIDです。

但し、テストIDを使用した場合は、右図のようなラベルが広告に表示され、クリック時にストアやLPページにではなく、弊社のサイトに遷移します。 また、インプも発生しません。

テストIDは、組み込み予定のアプリへの使用等、本サンプル以外で使用しても問題ありません。 ※ リリース時は、必ずテストIDから実際のIDに変更してからリリースしてください。

【テストID使用時の表示イメージ】



【各広告種類ごとのテストID】

package imobile.sample; public class SpotParams { // インタースティシャルスポット static final String INTERSTITIAL PUBLISHER ID = "34816": static final String INTERSTITIAL MEDIA ID = "135179": static final String INTERSTITIAL SPOT ID = "342418"; // ウォールスポット static final String WALL PUBLISHER ID = "34816"; static final String WALL_MEDIA_ID = "135179"; static final String WALL_SPOT_ID = "342419"; // スマートフォンバナースポット static final String BANNER_PUBLISHER_ID = "34816"; static final String BANNER MEDIA ID = "135179"; static final String BANNER_SPOT_ID = "342414"; // スマートフォンビッグバナースポット static final String BIG BANNER PUBLISHER ID = "34816": static final String BIG BANNER MEDIA ID = "135179"; static final String BIG BANNER SPOT ID = "342415"; // ミディアムレクタングルスポット static final String RECTANGLE_BANNER_PUBLISHER ID = "34816"; static final String RECTANGLE_BANNER_MEDIA_ID = "135179"; static final String RECTANGLE BANNER SPOT ID = "342416"; // アイコンスポット static final String ICON PUBLISHER ID = "34816"; static final String ICON MEDIA ID = "135179": static final String ICON_SPOT_ID = "342417"; // テキストポップアップスポット static final String TEXT POPUP PUBLISHER ID = "34816"; static final String TEXT POPUP MEDIA ID = "135179"; static final String TEXT_POPUP_SPOT_ID = "342420" ;

《全広告共通》 テストモード [sdкバージョン 2. 0. о以降]

テストモードを設定することにより、以下の動作を行う様になります。

・テストモードとして動作していることを示すラベルが表示される様になります。



エラー時にエラー内容が表示される様になります。
 【テストモード使用時のエラー表示イメージ】



エラー内容	失敗理由
AUTHORITY	設定されたスポット情報に誤りがあるため
NETWORK_NOT_READY	ネットワークが利用できないため
NETWORK	通信中にエラーが発生したため
NOT_DELIVERY_AD	配信できる広告がありません
RESPONSE	ネットワークのレスポンス不正のため
UNKNOWN	不明なエラーが発生したため ※頻発する場合は、ログおよび組み込み部分のソースを を添付し、弊社担当営業まで、お問い合わせください。
AD_NOT_READY	広告準備中のため広告の表示ができないため
SHOW_TIMEOUT	広告表示処理がタイムアウトによりキャンセル

・インプが送信されなくなります。(管理画面上のインプ数のカウントに含まれなくなります。)

適用した場合のみです

・クリック時に、弊社のサイトに遷移する様になります。(管理画面上のインプ数のカウントに含まれなくなります。)

【テストモードの設定方法】 「AndroidManifest.xml」の<application>セクション内に、テストモード用の以下の<meta-data>タブを追加します。 <meta-data android:name="i-mobile_Testing" android:value="true" /> ※ android:value="true" を android:value="false"にすることでテストモードを解除することができす。

《全広告共通》 ImobileSdkAdListener リスナーによる各種通知設定 (1/2)

ImobileSdkAdListenerをスポット、広告表示処理に設定することにより、さまざまな通知を受け取ることができます。

```
import ip.co.imobile.sdkads.android.ImobileSdkAdListener:
                                                                                                                      ······ 1
import jp.co.imobile.sdkads.android.FailNotificationReason;
@Override
protected void onCreate(Bundle savedInstanceState) {
  super.onCreate(savedInstanceState);
  setContentView(R.layout.activity detail);
   //広告の取得
   ImobileSdkAd.registerSpot(this, PUBLISHER_ID, MEDIA_ID, SPOT_ID);
   //スポットへのリスナーの設定
   ImobileSdkAd.set/mobileSdkAdListener(SPOT ID, new ImobileSdkAdListener() {
                                                                                                                                     0
       @Override
      public void onAdReadyCompleted() {
           //広告の表示準備が完了した際に呼び出されます
       @Override
      public void onAdShowCompleted() {
           //広告が表示された際に呼び出されます
       @Override
       public void onAdCliclkCompleted() {
           //広告がクリックされた際に呼び出されます
       @Override
      public void onAdCloseCompleted() {
           //広告が閉じられた際に呼び出されます
      }
       @Override
      public void onFailed(FailNotificationReason reason) {
          //広告の取得に失敗した際に呼び出されます。(reasonは、失敗理由が設定されます)
   });
   //広告の取得開始
   ImobileSdkAd.start(SPOT_ID);
```

- ① ImobileSdkAdListener と FailNotificationReason をインポートします。
- ② スポットにリスナーを設定します。(利用しない「on~」メソッドは省略可能です) ※リスナーは、広告表示処理(showAd)にも、設定可能です(設定方法は、「組み込み方法 【戻るボタン押下時に広告を表示する】」参照)が、通知される内容は限られます

レイアウトファイルにXMLにて組み込みを行っている場合はこの記述ができないため、リスナーを設定する場合はコードによる組み込みを行ってください。

《全広告共通》 ImobileSdkAdListener リスナーによる各種通知設定 (2/2)

ImobileSdkAdListenerが通知する内容を以下に示します。

メソッド名		通知内容	全画面広告の スポットに設定 した場合の通知有無	インライン広告 のスポットに 設定した場合 の通知有無	広告表示処理に設定 した場合の通知有無 ※全画面広告に対して のみ設定可能です
onAdReadyCompleted		広告の表示準備が完了しました	インタースティシャルの 場合のみ通知されます	通知されます	通知されません
onAdShow	Completed	広告の表示が完了しました	通知されます	通知されます	通知されません
onAdCliclk	Completed	広告がクリックされました	通知されます	通知されます	onAdCloseCompleted として通知されます
onAdClose	Completed	広告が閉じられました	通知されます	通知されません	通知されます
onFailed		広告の取得に失敗しました	通知されます	通知されます	通知されません
	失敗理由コード	失敗理由	通知されます	通知されます	通知されません
	AUTHORITY	設定されたスポット情報に誤りがあるため	通知されます	通知されます	通知されません
	NETWORK_NOT_READY	ネットワークが利用できないため	通知されます	通知されます	通知されません
	NETWORK	通信中にエラーが発生したため	通知されます	知されます	通知されません
	NOT_DELIVERY_AD	配信できる広告がありません	通知されます	通知されます	通知されません
	RESPONSE	ネットワークのレスポンス不正のため	通知されます	通知されます	通知されません
	UNKNOWN	不明なエラーが発生したため ※頻発する場合は、ログおよび組み込み部分のソースを を添付し、弊社担当営業まで、お問い合わせください。	通知されます	通知されます	通知されません
	AD_NOT_READY	広告準備中のため広告の表示ができないため	通知されます	通知されません	onAdCloseCompleted として通知されます
	SHOW_TIMEOUT	広告表示処理がタイムアウトによりキャンセル	通知されます	通知されません	onAdCloseCompleted として通知されます

《全広告共通》 バックグラウンド時に広告の取得を停止する

ImobileSdkAds.stopAll () を使用することにより、バックグラウンド時に広告の取得 (先読み処理) を停止することができます。 ※先読み処理を停止することで、バックグラウンド時の通信を止めることができますが、 復帰時にインタースティシャル広告を表示する様な組み込みをしている場合、インタースティシャル広告が表示できない場合があります。

import jp.co.imobile.sdkads.android.ImobileSdkAdListener; import jp.co.imobile.sdkads.android.FailNotificationReason;

@Override

protected void onCreate(Bundle savedInstanceState) {
 super.onCreate(savedInstanceState);
 setContentView(R.layout.activity detail);

//広告の取得

ImobileSdkAd.registerSpot(this, PUBLISHER_ID, MEDIA_ID, SPOT_ID);

//広告の取得開始

ImobileSdkAd.start(SPOT_ID);

J

}

@Override

protected void onPasue () { //広告の取得を停止 ImobileSdkAd.*stopAll();* ◀ super.onPause();

}

}

① バックグラウンドになった際に、広告の取得処理を停止します。

② バックグラウンドから復帰した際に、広告の取得処理を再開します。

レイアウトファイルにXMLにて組み込みを行っている場合でも、バックグラウンド時に広告の取得を停止させる場合は、この記述が必要になります。

--- 🕦

..... 🤈

■ 広告が表示されません

AndroidManifist.xml に以下のログ出力設定の記述を行い、ログを出力しエラーが発生していないかを確認してみてください。



■ インライン広告を一時的に非表示にしたい

以下のソースの様に広告を表示するViewGroupを指定し、ViewGroupを表示/非表示にすることにより実現可能です。

// 広告を表示するためのLayoutを追加
FrameLayout layoutAd = new FrameLayout(activity);
this.addContentView(layoutAd, 0,0);
// 広告を表示するためのLayoutに指定して広告を表示
ImobileSdkAd.showAd(DetailActivity.this,SpotParams."SPOT_ID", layoutAd);
// 広告を非表示(広告を表示するためのViewを非表示に設定)
layoutAd.setVisibility(View.GONE);
// 広告を再表示(広告を表示するためのViewを表示に設定)
layoutAd.setVisibility(View.VISIBLE);

初期の段階で広告が表示されない

広告が表示されない原因には以下が想定されます。

- ・アプリケーションの承認がi-mobileにて行われていない サイト/アプリ管理一覧に 審査中 アイコンが表示されている場合は未承認となります。
- i-mobile側にて配信するべき広告の選択が完了していない
 i-mobile側の処理が終わるまで、しばらく待ってからご確認ください。

| Google Play Services のインストール方法

 Android SDK Managerを開き、Google Play Servicesをインストールします。 Accept Licenceを選択し、Finishをクリックします。

② Package Explorerを右クリックし、インポートをクリックします。



「Existing Android Code Into Workspace」を選択し、[次へ] をクリックします。 [参照...]をクリックし、ライブラリプロジェクトを選択した後、 google-play-services_libにチェックが入っていることを確認し、 [完了]をクリックします。 ※ google-play-services_lib は以下のフォルダにインストールされています。 "ANDROID_HOME"/extras/google/google_play_services/libproject/

③ google-play-services_lib プロジェクトをインポートします。

■ Google Play Services のインストール方法

④ プロジェクトのプロパティを開き、[Android] を選択し、 Libraryセクションの [Add] をクリックします。

⑤ google-play-services_libを選択肢、[OK] をクリックします。

 ⑥ Libraryセクションに、google-play-services_libが追加されていることを 確認し、[OK] をクリックします。

以上でGoogle Play Services のインストールおよび、設定は完了です。

Google Mobile Ads SDKを利用している場合、Google Play Services をインストールする とクラスの多重定義でビルドエラーになります。 このエラーは、Google Mobile Ads SDKをプロジェクトから削除することで回避できます。





estV1_1 のプロパティー				
入力	Android		()	· •
- ス oid	Project Build Target			
oid Lint Preferences	クーゲットダ	11/11/1 =	[75:b [49]	- L
skatule	Android 1.5	Android Open Source Project	15 3	-
lewer ブラグイン	Google AP1s	Google Inc.	15 3	
Dugs	Android 1.5	Android Open Source Project	15 4	
1749-	Google AP1s	Goode Inc.	1.5 4	
11/2/35	Android 2.1	Android Open Source Project	2.1 7	
のビルド・パス	Google AP1s	Google Inc.	2.1 7	
doc ロケーション	Android 2.2	Android Open Source Project	2.2 8	
cet	Google APIs	Google Inc.	22 8	
ext 5- 0/1	Android 2.3.3	Android Open Source Project	233 10	
7・リポジトリー	Google APIs	Google Inc.	23.3 10	
9-	Android 3.0	Android Open Source Project	3.0 11	
ジェクト・ファセット	Google APIs	Google Inc.	3.0 11	
ジェクト参照	Android 0.1	Android Open Source Project	3.1 12	
/デバーグ設定	Google APts	Google Inc.	3.1 12	
	Android 3.2	Android Open Source Project	3.2 13	
	Google APIs	Google Inc.	3.2 13	
	Google TV Addon	Google Inc.	3.2 13	
	Android 4.0	Android Open Source Project	40 14	
	Google APIs	Google Inc.	4.0 14	
	Android 4.0.3	Android Open Source Project	4.0.3 15	
	Google APIs	Google Inc.	4.0.3 15	
	Android 4.1.2	Android Open Source Project	4.1.2 16	
	Google APIs	Google Inc.	4.1.2 16	
	Android 4.2.2	Android Open Source Project	42.2 17	
	Google AP1s	Google Inc.	42.2 17	
	Android 4.4.2	Android Open Source Project	4.4.2 19	
	Android + Google APIs			
	ライブラリー			
	T Is Library			
_				
	1	-TO CLODE	- DR BF	٦
	JJJJJUsers/Ad	ministr		=
<u> </u>			- m ±	
			L Ka	- I
			下へ	
				-
		0	-	
		(6)-	18元) 通用	6
			-	
			DK キャン・	z.n

■ Android 4.0.x、4.1.xにおいて、アイコン広告の背景が透過されない

Android Manifest にてアイコン広告を表示しているActivity に対して以下のパラメータを追加してください。

SharedPreferencesに保存している情報について

本SDKでは、以下の情報を SharedPreferencesを利用してデバイスに保存しています。

Кеу	Value	利用目的	弊社サーバへの 送信有無
"ImobileSdkAdsShowTimeOut" + パートナーID + メディアID + スポットID ※ 設定されているスポットの数だけ登録	広告表示処理(showAdメソッド)を呼び出してから、 広告表示を行う際のタイムアウト時間(秒)	広告表示処理(showAdメソッド)を呼び 出してから、広告表示をキャンセルする までの時間(秒)を制御するために使用	0
"ImobileSdkAdsTextPopupSkipCount" + パートナーID + メディアID + スポットID ※ 設定されているスポットの数だけ登録	管理画面にて設定されたテキストポップアップ広告の スキップ回数	テキストポップアップ広告の 表示スキップ制御に使用	0
"ImobileSdkAdsTextPopupSkipCounter" + パートナーID + メディアID + スポットID ※ 設定されているスポットの数だけ登録	広告表示処理(showAdメソッド)をスキップした回数 ※広告が表示された場合はリセット	テキストポップアップ広告の 表示スキップ制御に使用	0

■ SDKが取得している情報および送信している情報について

総務書が取りまとめを行っている、「スマートフォンプライバシーイニシアティブ」「スマートフォンプライバシーイニシアティブ II 」おいて、 アプリケーションが取得および送信している情報について、プライバシーボリシーを作成、公表するものとするという指針が提示されております。 参考URL:「スマートフォン等のアプリケーション配信事業者対象:利用者情報の取扱い、アプリケーション・プライバシーポリシーについて(JIPDEC)」 http://privacymark.jp/news/2014/0114/news_140114.pdf 「スマートフォン プライバシー イニシアティブ ー利用者情報の適正な取扱いとリテラシー向上による新時代イノベーションー」(総務省)」 http://www.soumu.go.jp/main_content/000171225.pdf 「スマートフォン プライバシー イニシアティブII ~アプリケーションの第三者検証の在り方~」(総務省) http://www.soumu.go.jp/main_content/000236366.pdf

アプリケーション、プライバシーポリシーにおいて、取得されている情報、および利用目的、第三者提供の有無等について記載にするべきとの指針が提示されております。 本SDKにおいても、適切かつ効果的に広告を配信するために、いくつかの情報を取得し、送信しておりますので、以下に取得項目について記載いたします。

取得、送信情報	概要	利用目的	弊社サーバへ の送信有無
パートナーID、メディアID、スポットID	i-mobileが発行しているID	配信する広告の適切な 選定および不正対策の ために利用	0
SDKのバージョン	SDKのバージョン		0
アプリケーションパッケージ名	広告を要求したアプリケーションパッケージ名		0
言語設定	デバイスの言語設定		0
0S	デバイスのOS		0
0Sバージョン	デバイスの0Sのバージョン		0
ネットワーク種別	デバイスの回線種別		0
デバイスブランド名	デバイスのブランド名(メーカー名)		0
デバイス名	デバイス名(機種名)		0
Google Advertising ID	AndroidOSに設定されている広告配信を目的とした識別子		0
画面解像度	画面解像度	配信する広告の適切な	0
画面サイズ(縦/横)	画面サイズ	選正のために利用	0
位置情報(緯度/経度)	現在の位置情報 ※位置情報を利用しているアプリケーションの場合のみ取得		0
インストール済パッケージ情報	デバイスにインストールされているパッケージ情報		×

弊社では、取得した情報につきまして提示した目的以外での利用および、 第三者への提供については一切行っておりません。