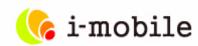
i-mobile for SP アプリ向け広告表示SDK iOS版 設定資料 AdWhirl対応



SDK バージョン 1.3

2012年04月版

目次

- 1. 概要
- 2. Ver1.2以前からアップデートするさいの注意点
- 3. AdWhirl対応
 - 1) SDKの設定
 - 2) 実装方法
 - 3) 実装サンプル

1. 概要

・はじめに

この資料はiPhone/iPadアプリに、AdWhirl SDK for iOSを利用してi-mobileの広告を表示する為のSDKの設定方法を記載した技術資料です。

通常の設定方法に関しては「i-mobile for SP アプリiOS設定資料.pdf」を参照ください。

・ 対象となるサービス



i-mobile for SP web

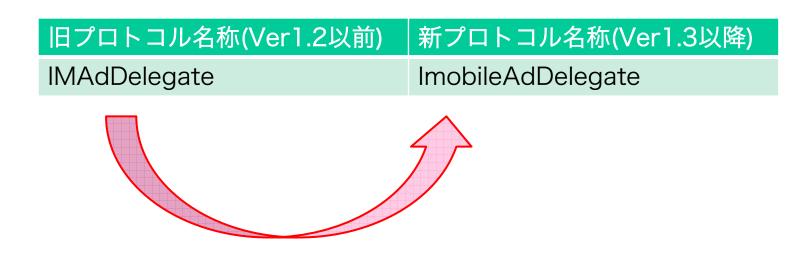
i-mobile for PC

i-mobile 携帯版

- ・対象となる読者
 - ・ AdWhirl SDKを利用してi-mobileの広告を表示する iPhone/ iPad アプリ開発者

2. Ver1.2以前からアップデートするさいの注意点

·i-mobile SDKはVersion 1.3からプロトコル名が変更になっています。 置換をお願いします。



3. AdWhirl対応 - 1)SDKの設定

1) SDKの設定

AdWhirlにて、i-mobileの広告を配信制御する手順を以下に記述します。

- ※ このドキュメントは、AdWhirl iOS SDK 3 に基づいて記述されています。
 - ① i-mobile SDKをアプリケーションのプロジェクトへ追加します
 - 「i-mobile for SP アプリ iOS 設定資料.pdf」を参照し、
 「5.1. XcodeによるSDKの設定」まで行います。
 - ② AdWhirl SDKをアプリケーションのプロジェクトへ追加します
 - AdWhirlの管理画面にて他社AdNetworkの設定をします。
 - AdWhirl SDKをアプリケーションのプロジェクトへ追加し、 広告が正しく表示されることを確認します。
 - ③ i-mobile AdNetworkをAdWhirlに登録します
 - AdWhirlの管理画面にてi-mobile AdNetworkを[CustomEvent]として追加します。 指定した[Function Name]をメモします。 サンプルでは、"handleimobileFunc"という名称で登録しています。 CustomEventの詳しい設定方法についてはAdWhirlのヘルプを参照ください。
 - AdWhirlの管理画面にて配信比率等を設定します。

※ 注意点

CustomEventとしてAdWhirl非対応のAdNetworkを複数登録した場合、 AdWhirl上のレポート表示は1つにまとめられてしまいます。

3. AdWhirl対応 - 2) 実装方法

2) 実装方法

以下の説明は「3.AdWhirl対応 - 1) SDKの設定」が完了していることを前提とします。

- IMobileAdDelegateプロトコルを実装します
 「1) SDKの設定② AdWhirl SDKをアプリケーションのプロジェクトへ追加します」にて、AdWhirlDelegateプロトコルを実装したViewControllerに
 IMobileAdDelegateプロトコルを追加し、以下の2メソッドをヘッダーに定義します。
 - (void)imAdViewDidFinishReceiveAd:(IMobileAdView*)imobileAdView;
 - (void)imAdViewDidFailToReceiveAd:(IMobileAdView*)imobileAdView;
- ② カスタムイベントの実装
 - 「1) SDKの設定 ③i-mobile AdNetworkをAdWhirlに登録します」にて登録したカスタムイベントを定義、実装します。
 - i-mobileの広告表示ビューを生成します [IMAdWhirlBannerView imAdWhirlBannerViewWithFrame:withDelegate:]メソッドを使用します。 withDelegateには、①で実装したIMobileAdDelegateプロトコルを指定します。 ※IMAdViewのinit系メソッドを用いてインスタンスを生成しないでください。
 - 広告表示に必要なIDを登録します
 [IMobileAdView setWithPublisherId: mediald: spotId:]メソッドを使用します。
 管理画面のアプリ設定値で発行された、パブリッシャーID、メディアID、スポットIDを指定します。
 - 広告の取得を開始します [IMobileAdView start]メソッドを使用します。
- ③ 広告取得|成功時の動作を実装します [IMobileAdDelegate imAdViewDidFinishReceiveAd:] メソッドを使用します。 [AdWhirlView replaceBannerViewWith:]メソッドを使用し、広告を切り替えます。
- ④ 広告取得|失敗時の動作を実装します [IMobileAdDelegate imAdViewDidFailToReceiveAd:] メソッドを使用します。 [AdWhirlView rollOver]メソッドを使用し、広告を切り替えます。

以上の手順を行うことで、i-mobileから配信される広告が表示可能になります。

3. AdWhirl対応 - 3)実装サンプル 1

3) 実装サンプル 1

新規に作成した「Single View Application」プロジェクトに実装する場合を例に説明します。

- ① 新規に「Single View Application」プロジェクトを作成します。
- ②「1) SDKの設定」の手順を実行します。 AdWhilrの[CustomEvent] には[Function Name]を"handleimobileFunc"として登録します。 AdWhirlDelegateプロトコルは、ViewControllerに追加します。
- ③ ViewControllerにIMobileAdDelegateプロトコルを追加します。 "ViewController.h"を以下のように編集します。

3. AdWhirl対応 - 3)実装サンプル 2

前ページから

3) 実装サンプル 2



- ④ ViewControllerにカスタムイベント等を追加します。"ViewController.m"を以下のように修正します。
 - ※ " MY AD WHIRL APPLICATION KEY"にはAdWhirl管理画面上のSDK KEYを指定します。
 - ※ "IMADVIEW_PUBLISHERID","IMADVIEW_MEDIAID","IMADVIEW_SPOTID"には i-mobile管理画面の「アプリ設定取得」で発行された値をそれぞれ指定します。

```
#import "ViewController.h"
@implementation ViewController
        ~ 略 ~
- (void)viewDidLoad {
  [super viewDidLoad];
  adWhirlView = [AdWhirlView requestAdWhirlViewWithDelegate:self];
  [self.view addSubview:adWhirlView]:
- (NSString *)adWhirlApplicationKey {
  return MY AD WHIRL APPLICATION KEY:
- (UIViewController *)viewControllerForPresentingModalView {
  return self:
- (void)handleimobileFunc:(AdWhirlView *)awView {
  IMAdWhirlBannerView *imobileAdWhirlBannerView =
     IIMAdWhirlBannerView imAdWhirlBannerViewWithFrame;kAdWhirlViewDefaultFrame
       withDelegate:self1:
  [imobileAdWhirlBannerView setWithPublisherId:IMADVIEW PUBLISHERID
       mediald:IMADVIEW MEDIAID
       spotId:IMADVIEW SPOTID];
  [imobileAdWhirlBannerView start];
- (void)imAdViewDidFinishReceiveAd:(IMobileAdView *)imobileAdView {
  [adWhirlView replaceBannerViewWith:imobileAdView]:

    - (void)imAdViewDidFailToReceiveAd:(IMobileAdView *)imobileAdView {

  [adWhirlView rollOver];
@end
```