

Freestar Ads Mediation Cordova iOS

Last Modified on 05/30/2022 5:09 pm EDT

Freestar Ads Mediation - Cordova Integration Guide for iOS

Supported Ad Partners

Note: The supported list for Cordova may differ from our Native iOS

Ad Provider	SDK Version	Ad Unit Types
AdColony	4.7.2	Fullscreen Interstitial & Rewarded
AppLovin	10.1.1	Fullscreen Interstitial & Rewarded, Banner 300x250, Banner 320x50
Admob	8.12.0	Fullscreen Interstitial & Rewarded, Banner 300x250, Banner 320x50, Native
Google Ads Manager	8.12.0	Fullscreen Interstitial & Rewarded, Banner 300x250, Banner 320x50, Native
MoPub (deprecated)	5.18.2	Fullscreen Interstitial & Rewarded, Banner 300x250, Banner 320x50, Native
Tapjoy	12.8.0	Fullscreen Interstitial & Rewarded
Unity Ads	3.7.2	Fullscreen Interstitial & Rewarded, Banner 320x50
Vungle	6.10.3	Fullscreen Interstitial & Rewarded
TAM (Amazon Publisher Services)	4.0.0	Fullscreen Interstitial, Banner 300x250, Banner 320x50
Pangle	3.7.0.8	Fullscreen Interstitial & Rewarded, Banner 300x250, Banner 320x50, Native
Hyprmx	6.0.0	Fullscreen Interstitial & Rewarded, Banner 300x250, Banner 320x50

Getting Started

Apache Cordova is a metaplatform for developing mobile apps with Javascript. This document will instruct you how to deliver Freestar Ads in mobile apps built with Cordova. Start displaying Freestar Ads in your Cordova app today by following the simple steps below.

Requirements

- NPM 6 or newer
- Cordova 9 or newer
- iOS SDK with Xcode for iOS 10 or higher
- Cocoapods

Begin

Please follow the instructions in order.

Go to your cordova app directory in a terminal and add the Freestar Cordova plugin:

```
$ cordova plugin add @freestar/cordova-plugin-freestar
```

If you do not wish to integrate every advertising partner, edit the following section of the `node_modules/cordova-plugin-freestar/plugin.xml` file so that only the partners you wish to include are listed:

```
<pods use-frameworks="true">
  <pod name="FreestarAds" spec="~> 5.3" />
  <pod name="FreestarAds-AdColony" spec="~> 5.1" />
  <pod name="FreestarAds-AppLovin" spec="~> 5.1" />
  <pod name="FreestarAds-Googleadmob" spec="~> 6.1" />
  <pod name="FreestarAds-Tapjoy" spec="~> 5.0" />
  <pod name="FreestarAds-Unity" spec="~> 6.0" />
  <pod name="FreestarAds-Vungle" spec="~> 5.2" />
  <pod name="FreestarAds-GAM" spec="~> 5.1" />
  <pod name="FreestarAds-TAM" spec="~> 2.2" />
  <pod name="FreestarAds-Pangle" spec="~> 1.1" />
  <pod name="FreestarAds-Hyprmx" spec="~> 1.0" />
  <pod name="FreestarAds-Mopub" spec="~> 5.1" /> (deprecated)
</pods>
```

Freestar Ads require an API key to deliver ads. Once you obtain the key, make sure it is added to the app's `Info.plist` file in the generated iOS app. You can either edit the plist directly after the app is generated, or modify the Cordova app's `config.xml` file:

```
<platform name="ios">
  <edit-config target="com.freestar.ios.ads.API_KEY" file="*-Info.plist" mode="merge">
    <string>YOUR_API_KEY</string>
  </edit-config>
</platform>
```

If, in the list of partners describes above, you've asked to include either `FreestarAds-Googleadmob` or `FreestarAds-GAM` pods, you will also need to modify the generated app's `Info.plist` file to include Google's ad API key:

```
<edit-config target="GADApplicationIdentifier" file="*-Info.plist" mode="merge">
  <string>YOUR_ADMOB_KEY</string>
</edit-config>
```

Once all the changes are made, you can generate the iOS app from your cordova codebase:

```
$ cordova platform add ios
```

Opening the `.xcworkspace` file in the generated project with Xcode will allow you to run the app on the iOS simulator or device.

Displaying Ads

If the Freestar Cordova plugin was successfully integrated, its apis will be accessible via the `window.plugins.freestarPlugin` object.

Interstitial Ads

Interstitial ads take over the app, running in fullscreen, and allow return to the app upon completion. You can opt in to receive callbacks when key events occur in an ad's lifecycle, allowing you to make any changes necessary (such as pausing your game while the ad is playing). To do this, your Cordova code needs to add event listeners to the corresponding ad events:

```
document.addEventListener('onInterstitialLoaded', onInterstitialLoaded, false);
document.addEventListener('onInterstitialFailed', onInterstitialFailed, false);
document.addEventListener('onInterstitialShown', onInterstitialShown, false);
document.addEventListener('onInterstitialClicked', onInterstitialClicked, false);
document.addEventListener('onInterstitialDismissed', onInterstitialDismissed, false);
```

Then, you request to load an ad:

```
window.plugins.freestarPlugin.loadInterstitialAd(AD_PLACEMENT);
```

Note: there are additional details of what placements are and the format they need to be in our [guide for native app integration](#). If you do not wish to use placements, simply pass in null:

```
window.plugins.freestarPlugin.loadInterstitialAd(null);
```

While implementing callbacks is optional, showing an ad before it's loaded will not work. The `onInterstitialAdLoaded` callback informs your Cordova code when the ad is ready. You can now make the following call:

```
window.plugins.freestarPlugin.showInterstitialAd(AD_PLACEMENT);
```

Make sure the placement strings match, if you are using more than one placement. The easiest way to do this is to show the ad from within the `onInterstitialAdLoaded` callback you added as an event listener for that event:

```
function onInterstitialLoaded(data) {  
    window.plugins.freestarPlugin.showInterstitialAd(data.placement);  
}
```

The ad will now take over the app, and make the appropriate callbacks; the `onInterstitialAdDismissed` is sent when the ad returns control back to the app.

Rewarded Ads

Rewarded ads behave like interstitial ads, but are augmented with an incentivization mechanism. Typically, the user is rewarded by in-app virtual currency upon completion of the ad. The currency and amount are under your control, and thus need to be passed to `window.plugins.freestarPlugin`.

To do this, your Cordova code needs to add event listeners to the rewarded ad events:

```
document.addEventListener('onRewardedVideoLoaded', onRewardedVideoLoaded, false);  
document.addEventListener('onRewardedVideoFailed', onRewardedVideoFailed, false);  
document.addEventListener('onRewardedVideoShown', onRewardedVideoShown, false);  
document.addEventListener('onRewardedVideoShownError', onRewardedVideoShownError, false);  
document.addEventListener('onRewardedVideoCompleted', onRewardedVideoCompleted, false);  
document.addEventListener('onRewardedVideoDismissed', onRewardedVideoDismissed, false);
```

And then request to load an ad:

```
window.plugins.freestarPlugin.loadRewardedAd(null);
```

While implementing callbacks is optional, showing an ad before it's loaded will not work. The `onRewardedVideoLoaded` callback informs your Cordova code when the ad is ready. You can now make the following call:

```
window.plugins.freestarPlugin.showRewardedAd(  
    AD_PLACEMENT,  
    "MySecret1234", //secret key  
    "MyUserId",    //user id  
    "Gold Coins",  //reward name  
    "100"          //reward amount  
);
```

Make sure the placement strings match, if you are using more than one placement. The ad will now take over the app, and make the appropriate callbacks; the `onRewardedVideoDismissed` is sent when the ad returns control back to the app.

The `onRewardedVideoCompleted` is called *before* the dismissed callback, and may be called when the ad is still playing. It is an indicator that the reward for the ad may now be given to the app user.

Banner Ads

Banner ads are displayed on top of the app, and only obscure it partially, rather than taking over the screen entirely. Freestar allows you to display either small (320x50) or medium (300x250) banners, and to choose to show them at the top, bottom, or middle of the device screen. You can also receive callbacks for key events in the ad's lifecycle.

To do this, your Cordova code needs to add event listeners to the rewarded ad events:

```
document.addEventListener('onBannerAdFailed', onBannerAdFailed, false);
document.addEventListener('onBannerAdShown', onBannerAdShown, false);
document.addEventListener('onBannerAdClicked', onBannerAdClicked, false);
document.addEventListener('onBannerAdDismissed', onBannerAdDismissed, false);
```

And then request the ad. There are no separate load and show calls:

```
window.plugins.freestarPlugin.showBannerAd(
  null, //the placement string -- null if you don't use multiple placements
  window.plugins.freestarPlugin.BANNER_AD_SIZE_320x50, //banner size
  window.plugins.freestarPlugin.BANNER_AD_SIZE_320x50 //banner position
);
```

The ad will load and display automatically at the appropriate position. When you no longer need it, you can remove it:

```
window.plugins.freestarPlugin.closeBannerAd(null, window.plugins.freestarPlugin.BANNER_AD_SIZE_300x250);
```

Make sure the placement string and size you pass in match.

Working sample

Take a look at the [sample project](#) and specifically its `www/index.js` file.

This working sample shows how to initialize, implement the ad listeners, and show fullscreen Interstitial, Rewarded, and banner ads.

Important iOS 14 Changes

One thing you must do as a publisher is set the Facebook `setAdvertiserTrackingEnabled` flag

appropriately, since Facebook is one of our partners. [Here is the official document from Facebook.](#) Generally speaking, one typical you could approach this is if the user has granted the app permission to track them at the OS level, then you would set this flag to true. By default, the flag will be false, meaning that Facebook will not serve any ads. We leave this implementation detail to you in order to give you, the app publisher, more control.

GDPR Update

We are releasing a major update to the Freestar SDK later this week that will require publishers to utilize a GDPR TCF 2.0 compliant CMP in order to render ads to users in a GDPR affected country. If the publisher is not utilizing a GDPR TCF 2.0 compliant CMP once upgraded to this SDK version, then ads will not serve to users in a GDPR affected country.

We put together a comprehensive list of FAQs below to help break down how the changes may impact your business. If you have additional questions, please reach out to your dedicated Account Manager and we will be happy to help on a case by case basis.

What is the new SDK version?

4.0.0 for iOS and Android

How do I become GDPR compliant?

In order to be able to serve ads to your users who reside in any EU country, you will need to implement an IAB TCF 2.0 CMP service.

The CMP service is essentially a personal data and privacy form that must be presented to users to collect their consent or dissent. This form can be presented as often as you like throughout the session of your game or app, as you may require your users to see ads. The most typical setup would be for the first visit to an App with an option to edit your preferences through some other call to action.

What if I already have a CMP implemented?

If you already have a CMP implemented, you will need to add Publisher First, Inc. (Freestar's official legal entity) as a vendor, as well as each of our partners listed here if not already included.

You will need to retrigger the consent form for GDPR affected users once Freestar and additional vendors are added to the vendor list.

If the list of vendors has already been included, no additional effort is required.

How does the Freestar SDK work with a CMP?

Freestar SDK will automatically detect the user's CMP response and act accordingly. More specifically, if the user consents, then Freestar SDK will be allowed to serve ads. If the user dissents, then Freestar SDK will not show ads.

What if I choose not to implement a CMP?

If a CMP is not implemented at all, then Freestar SDK will not show ads to users who reside in a GDPR affected country.

What do I need to do once I choose a CMP?

After you have chosen a CMP Service provider, during configuration and setup of your CMP, you will need to include a list of supported vendors. We recommend that you select 'Include All Vendors' rather than choosing specific vendors, as your list of vendors may change over time.

If you choose to add vendors to your list manually, you will need to include Publisher First, Inc. (Freestar's official legal entity) as a vendor as well as Freestar's list of supported vendors here.

Does Great Britain still observe GDPR requirements despite leaving the E.U.?

Yes, it does still apply. In anticipation of Brexit, a new domestic data privacy law called the UK-GDPR took effect on 1/31/20. The UK-GDPR is almost word for word completely identical to the EU's GDPR.

CMP Recommendations:

Consent Manager - <https://www.consentmanager.net/> App Consent - <https://sfbx.io/en/produits/>

Here is a list of IAB approved CMP Service providers you can implement in your game or app: <https://iab europe.eu/cmp-list/>

Testing

For iOS, please use our iOS test key **91784edd-3492-4111-8742-f71bd3803dd3** (we have a different key for Android, so don't use for both iOS and Android) for all your iOS testing runs and enable test mode.

Turn on test mode:

```
FSTR_TEST_ADS
```

You will usually get 100% fill on all ad units.

It is not recommended to use your production key for testing runs as that is strictly prohibited by our partners and bad things may happen to us on the business side of things.

Do not forget to uninstall and re-install your app when changing keys on your device.

When you are satisfied with your testing, please make a release build with your production key, and turn test mode off. Publish to store.

SKAdNetwork IDs

iOS 14 changed the way advertising works on iOS devices. Ad effectiveness tracking requires the usage of `SKAdNetwork` APIs. To enable this, `SKAdNetwork` keys should be included in your app's `Info.plist` file.

Please see our list: https://github.com/freestarcapital/SDK_documentation_iOS/wiki/iOS-14-SKAdNetwork-IDs

M1 Macs

Currently, we do not support the building of our SDK on M1 macs. This will be addressed in an upcoming release.

Xcode 12 related cocoapods build errors

Recently, in Xcode 12.2, Apple made some breaking changes in Xcode related to build settings. By default, arm64 architecture is now added to ARCHS_STANDARD. In addition, they have removed support for VALID_ARCHS setting from Xcode. Apple also added, in Xcode 12, a new build setting called **Excluded Architectures**. The reason Apple added arm64 is to support simulator on M1 Mac hardware. However this causes build issues running simulator on Intel Macs. As a result, publishers running Xcode on Intel Macs will need to make usage of the EXCLUDED_ARCHS build setting, for simulator testing, to exclude the arm64 simulator. So, if your Xcode build is failing with arm64 simulator errors in the log, such as:

```
building for iOS Simulator, but linking in object file built for iOS, file for architecture arm64
```

then, it is suggested to use this cocoapods post install hook in your Podfile:

```
post_install do |installer|
  installer.pods_project.build_configurations.each do |config|
    config.build_settings["EXCLUDED_ARCHS[sdk=iphonesimulator*]"] = "arm64"
  end
end
```

This is a temporary fix and should address the issue of build failures resulting from Xcode attempting to build (arm64) simulator on Intel Macs. As of our latest SDK release, we do this automatically in the podspec via xcconfigs, however not every SDK vendor has adopted this workaround. This post install hook is only needed if you are running Xcode 12 on Intel hardware and wanting to run simulator, and you have a cocoapods dependency that is not Xcode 12 / arm64 compliant.

Firebase Dependency

If your app integrated the Firebase SDK, reference the Firebase version compatible with current

release 7.0

```
pod 'Firebase/Core', '~> 7.0'
```

For further reference:

<https://github.com/CocoaPods/CocoaPods/issues/10104>

<https://stackoverflow.com/questions/63607158/xcode-12-building-for-ios-simulator-but-linking-in-object-file-built-for-ios/63955114#63955114>

Yahoo Demand Partner Configuration

If Yahoo is being added as a demand partner in your Podfile, then it is required to add an entry into your Info.plist. See example below:

```
<key>VerizonAdsSourceAppId</key>  
<string>Replace_this_with_your_App's_App_Store_ID</string>
```

GAM and Googleadmob OB adapters

If you would like to make usage of Google Open Bidding (OB) adapters for FreestarAds, please contact your account manager (AM) for further instructions. We do not advise including OB adapters into your project without first consulting with your AM.