

# Freestar Ads Mediation for Unity Android

Last Modified on 05/31/2022 12:13 pm EDT

Freestar Ads Mediation - Unity Integration Guide for Android

## Getting Started

Start displaying Freestar Ads in your Unity game today by following the simple steps below.

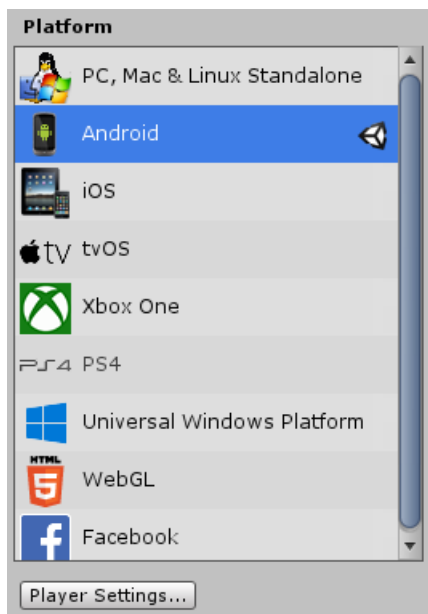
## Requirements

- Unity 2018 or newer
- Minimum Android API Level: 21 or higher

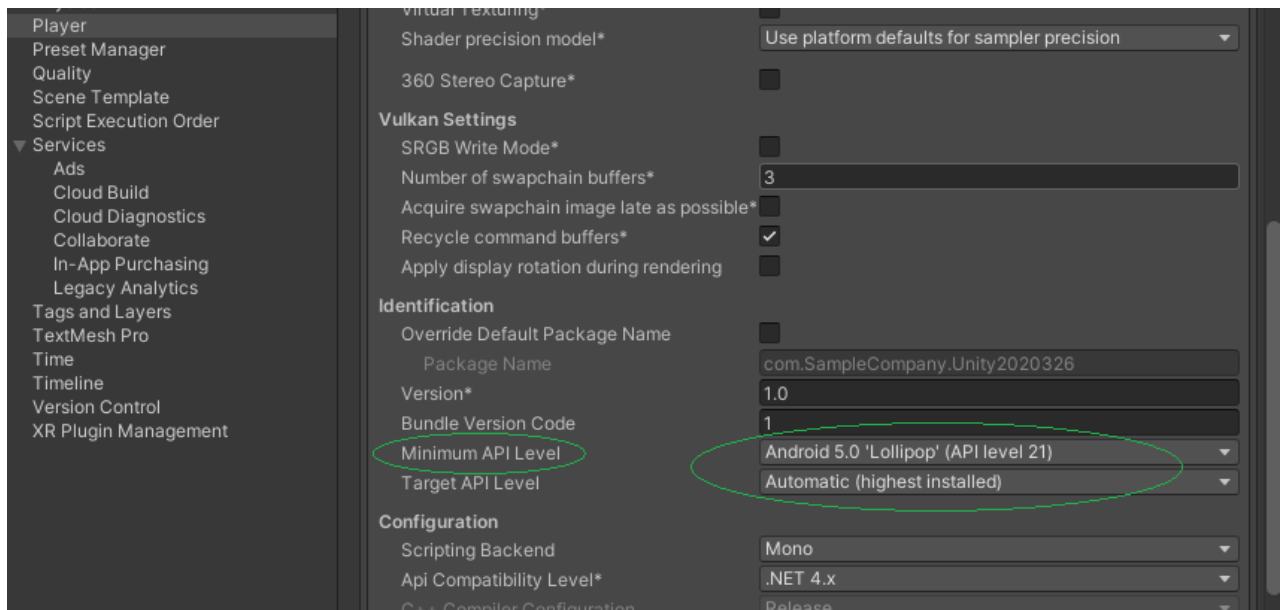
## Begin

Please follow the instructions in order.

Make sure you switched to Android:



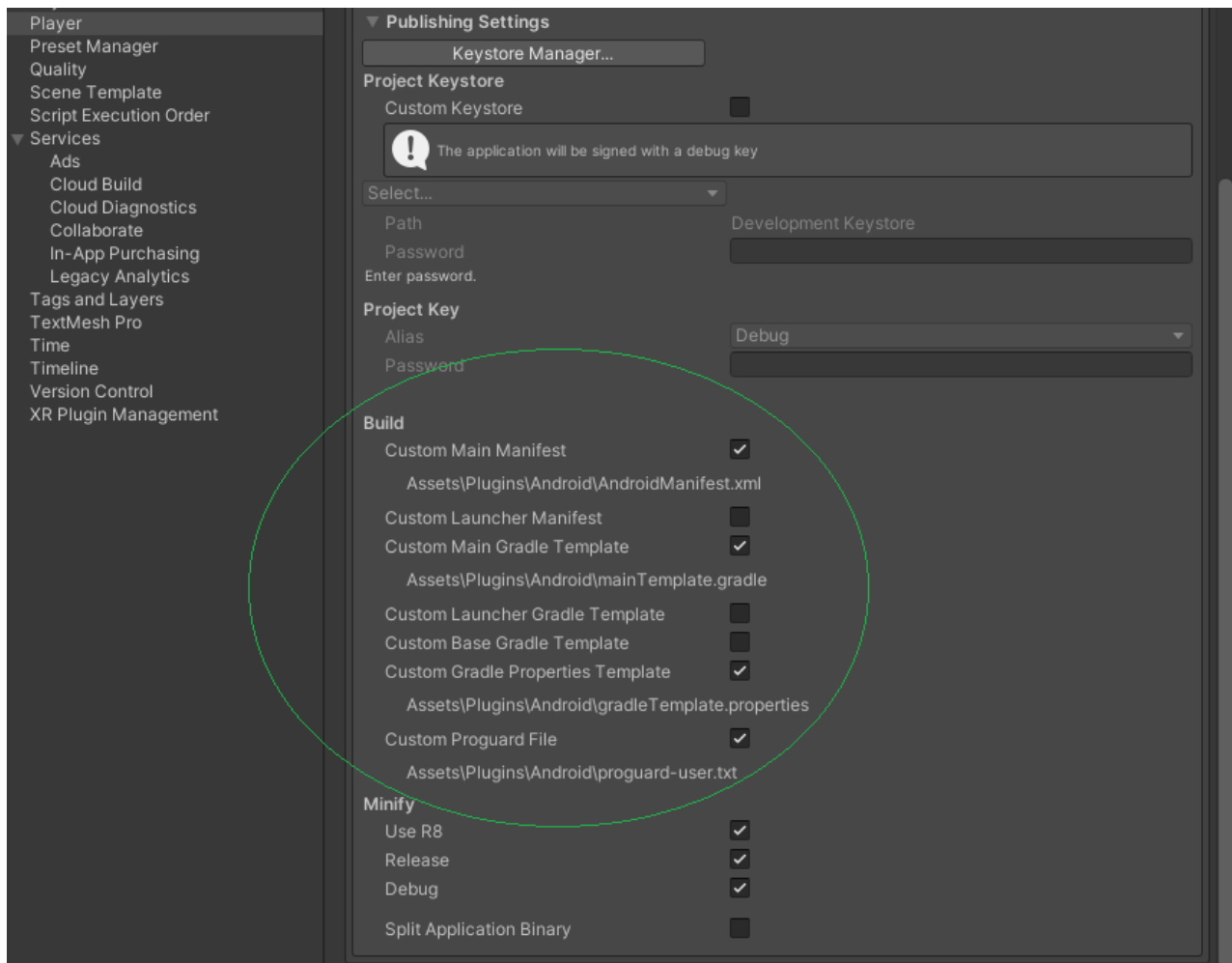
## Unity Player Settings



Other settings: Minimum API Level: 21

Other settings: Target API Level: Highest Installed

Other settings: Api Compatability Level: .NET 4.x



Publisher Settings: Custom Gradle Template □

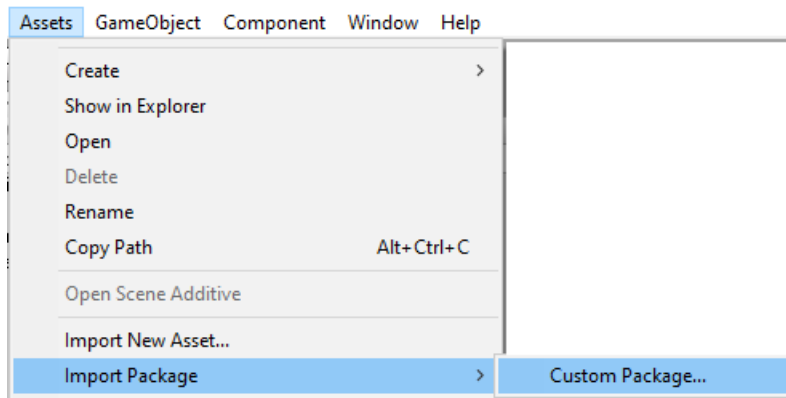
## Do Not Install Unity Ads

If you try to use Unity Ads Package with FreestarMediation, this will produce a build-time error because FreestarMediation already contains the Unity Ads library.

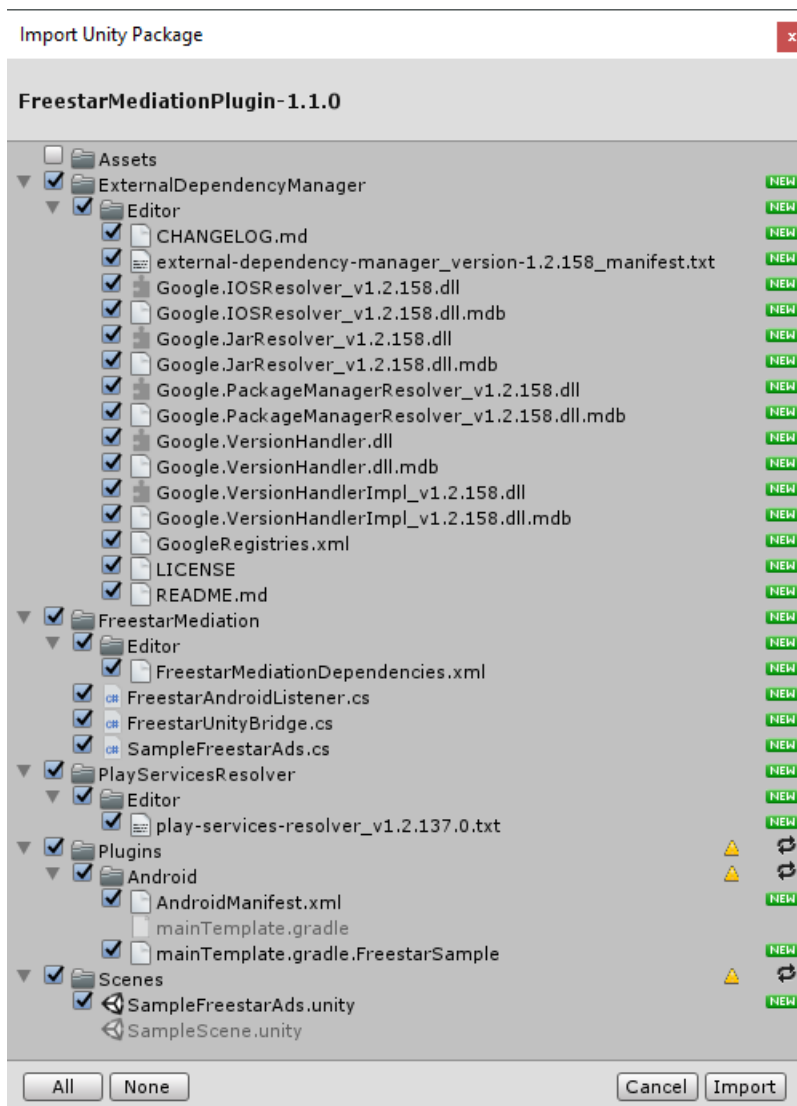
## Import Custom Package

Download [FreestarMediation.unitypackage](#)

Next, we will import it into Unity as a Custom Package.



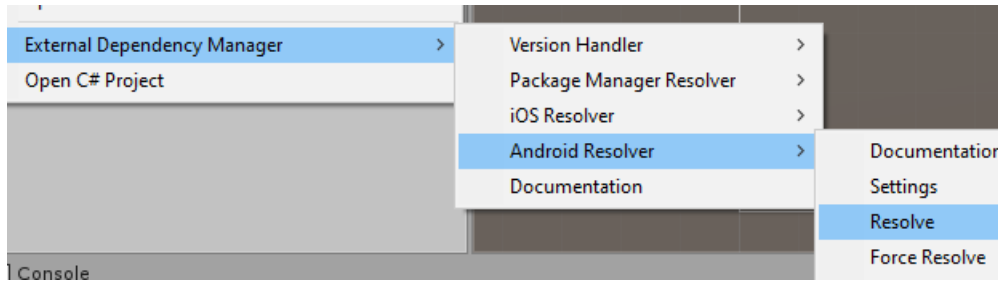
## Assets: Import Package: Custom Package



Hit the Import button.

# Resolve Android Dependencies

After importing our custom package, resolve Android dependencies via Assets.. External Dependency Manager... Android Resolver... Resolve.



## Manually Edit mainTemplate.gradle

Use your favorite file editor.

Edit [Unity Project Folder]/Assets/Plugins/Android/mainTemplate.gradle

Remove the top line:

```
// GENERATED BY UNITY. REMOVE THIS COMMENT TO PREVENT OVERWRITING WHEN EXPORTING AGAIN
```

```
// Android Resolver Exclusions End
android {
    compileSdkVersion **APIVERSION**
    buildToolsVersion '**BUILDTOOLS**'

    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }

    packagingOptions {
        exclude("META-INF/*.kotlin_module")
    }

    defaultConfig {
        minSdkVersion **MINSDKVERSION**
        targetSdkVersion **TARGETSDKVERSION**
        ndk {
            abiFilters **ABIFILTERS**
        }
        versionCode **VERSIONCODE**
        versionName '**VERSIONNAME**'
    }
}
```

Add the following to the android root section:

```
packagingOptions {
    exclude("META-INF/*.kotlin_module")
}
```

Save your changes to mainTemplate.gradle

## AndroidManifest.xml

Take a look at file [Unity Project Folder]/Assets/Plugins/Android/AndroidManifest.xml

Before going to production, you will need to modify the following values. For now, you can use what's already there.

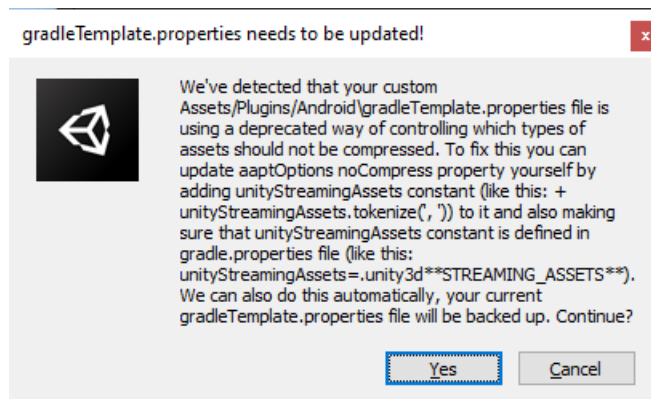
```
<meta-data
    android:name="com.google.android.gms.ads.APPLICATION_ID"
    android:value="ca-app-pub-3940256099942544~3347511713"/>

<meta-data
    android:name="applovin.sdk.key"
    android:value="hO52kFtMvEo_AoeRzED0_XXfS1B1VQp9GW50yudJO-eUUTOmRBLI3c-2GyTevLNspI_fN5PLtBAHO
akoTuHuP" />

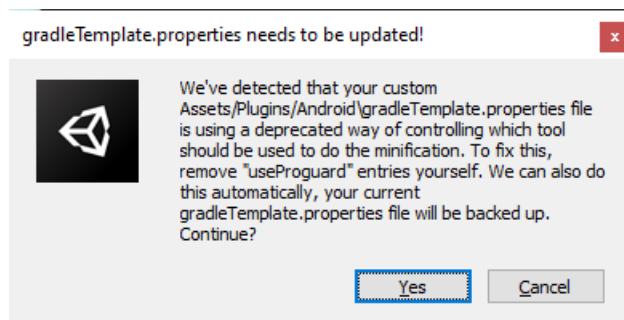
<meta-data android:name="com.google.android.gms.version" android:value="@integer/google_play_services_
version"/>
```

## Ready to Build Android

At this point, you should be able to build Android. If you encounter the following dialogs, hit 'YES'.



OR



## Freestar Initialization and Displaying Ads

Take a look at `Assets/FreestarMediation/SampleFreestarAds.cs` which was imported into your Unity project.

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

`Assets/FreestarMediation/SampleFreestar.scene` is also included.

Some publishers like to create a singleton class based on this sample. Utilize it as you see fit.

You can use the test key in `SampleFreestarAds.cs`

## Privacy - Google Play Families Policy Compliance

If your game or app is officially under the [Google Play Families](#) program, Freestar provides such support:

```
FreeStarAds.setGoogleFamilyPolicyMode( GoogleFamilyPolicyMode.app, true); //If your app is designed only for children
//FreeStarAds.setGoogleFamilyPolicyMode( GoogleFamilyPolicyMode.mixed, false); //If your app is designed for families with children
FreeStarAds.init(...);
```

If your app is not officially under the Google Play Families program, then you do not need to set the Google Family Policy mode.

```

/**
 * Only set Google Families Policy Mode if your app is required to comply with Google Play's
 * Families Policy program.
 *
 * @param googleFamilyPolicyMode app: the app is child-directed and will not receive
 *         personalize or contextual ads.
 *         mixed: the app is directed at mixed audiences.
 *         none: (default) the app is not required to comply with Google
 *         Play's Family Policy
 *
 * @param onlyNonPersonalizedAds true: if 'mixed' mode, then only personalized or contextual ads
 *         may be served.
 *         false: if 'mixed' mode, then personalized or contextual ads
 *         may be served.
 *         note: if 'app' mode, then personalized or contextual ads
 *         may not be served regardless of this parameter.
 */
public static void setGoogleFamilyPolicyMode(GoogleFamilyPolicyMode googleFamilyPolicyMode,
        boolean onlyNonPersonalizedAds)

```

## GDPR

Freestar SDK is GDPR compliant. In order for your users to be able to receive any ad fills in GDPR-affected countries, you, as a publisher, will need to implement a 3rd party Consent Management Platform (CMP). Freestar SDK will automatically detect user interactions with the CMP and act accordingly. Please see our [Freestar GDPR Frequently Asked Questions](#) for complete details and our recommended list of CMP service providers.

## Testing

For Android, please use our test key **XqjhRR** for all your testing runs and enable test mode true. 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.

## Automated Testing - Bypassing Ads

Are your automated tests failing after integrating Freestar Mediation Ads into your mobile application or game? Are you not sure it could be due to Freestar or something else? We have a feature called **Automated Test Mode** where you can run your automated tests to bypass Freestar or run Freestar in 'Limited Mediation' mode without making drastic changes to your code:

In your automated test suite code, **before** `FreeStarAds.init` is called:



```
FreeStarAds.setAutomatedTestMode( FreeStarAds.AutomatedTestMode.BYPASS_ALL_ADS )  
//OR  
FreeStarAds.setAutomatedTestMode( FreeStarAds.AutomatedTestMode.LIMITED_MEDIATION ) //only runs AdMob &  
GAM
```

Again, this is only for your automated tests and not for production use.

## Release Build

Final note when creating your release build: if you use [proguard](#) or [minify](#), please add our [proguard rules](#) to your app proguard file.