

# Android SDK Integration Guide

This document details the process of integrating the Beachfront .iO Ad SDK with your Android application.

## Requirements

---

1. Beachfront .iO app ID & Ad Unit ID - [Get it from here](#).
2. Beachfront SDK Jar - [Get it from here](#).
3. Android 1.5 and above or Google TV

## Installation

---

1. Access the Beachfront.iO Console & register your application to get your App ID & Ad unit ID;
2. Download the Beachfront Android AD SDK & Android Support Library, copy into the /lib folder of your Android Project. Make sure the jar files in lib is selected in the Order and Export tab of the Java Build Path panel.
3. Add the BF activity in the application node of AndroidManifest.xml:

```
<activity
    android:name="com.bfio.ad.BFIOActivity"
    android:configChanges="keyboardHidden|orientation|screenSize"
/>
```

Make sure to have android:targetSdkVersion should be equal or greater then 13 in the manifest.

```
<uses-sdk android:targetSdkVersion="13" />
```

Add following required permission in your AndroidManifest.xml

```
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

## Showing Interstitial Ads

---

Wherever you want to show an interstitial ad, declare a `BFIOInterstitial` instance variable, register your activity as the interstitial's `BFIOInterstitial.InterstitialListener`, and instantiate the instance in the `onCreate(Bundle savedInstanceState)` method, as shown below:

```
public class MainActivity extends Activity implements
BFIOInterstitial.InterstitialListener {

    BFIOInterstitial interstitial;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        interstitial = new BFIOInterstitial(MainActivity.this, this);
    }
}
```

Your application is now ready to display an interstitial ad.

Call `requestInterstitial` method to request an interstitial ad. You have to pass the `appId` & `adUnitId`

```
interstitial.requestInterstitial("appId", // appId    "adUnitId"); //adUnitId
```

In case Ad Server find a AD it will return to

```
public void onReceiveInterstitial(BFIOInterstitialAd ad);
```

Otherwise will return to:

```
public void onInterstitialFailed(BFIOErrorCode errorCode);
```

To start the Interstitial Call the following method

```
interstitial.showInterstitial(ad);
```

Following are callbacks methods of an interstitial ad:

```
/**
 * On Interstitial Failed
 *
 * @param errorCode
 */
public void onInterstitialFailed(BFIOErrorCode errorCode);

/**
 *Interstitial Displaying on the screen
 *
 */
public void onInterstitialStarted();

/**
 * Interstitial clicked
 *
 */
public void onInterstitialClicked();

/**
 * Interstitial dismissed
 *
 */
public void onInterstitialDismissed();

/**
 * Interstitial completed
 *
 */
public void onInterstitialCompleted();

/**
 * On Interstitial Received
 *
 */
public void onReceiveInterstitial(BFIOInterstitialAd ad);
```

Example Code:

```
public class MainActivity extends Activity implements
    BFIOInterstitial.InterstitialListener, OnClickListener {

    BFIOInterstitial interstitial;
```

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    findViewById(R.layout.activity_main);
    interstitial = new BFIOInterstitial(MainActivity.this, this);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

@Override
protected void onDestroy() {
    interstitial.onDestroy();
    super.onDestroy();
}

@Override
protected void onInterstitialFailed(BFIOErrorCode errorCode) {
    Toast.makeText(MainActivity.this, "Interstitial not Received",
        Toast.LENGTH_SHORT).show();
}

@Override
public void onInterstitialClicked() {
    Toast.makeText(MainActivity.this, "Interstitial Clicked",
        Toast.LENGTH_SHORT).show();
}

@Override
public void onInterstitialDismissed() {
    Toast.makeText(MainActivity.this, "Interstitial dismissed",
        Toast.LENGTH_SHORT).show();
}

@Override
public void onReceiveInterstitial(BFIOInterstitialAd ad) {
    Toast.makeText(MainActivity.this, "Received Interstitial",
        Toast.LENGTH_SHORT).show();
    interstitial.showInterstitial(ad);
}

@Override
public void onClick(View v) {
    switch (v.getId()){
```

```

        case R.id.request:
            interstitial.requestInterstitial("appid", // appId
                                           "addUnitId"); // addUnitId
            Toast.makeText(MainActivity.this, "Interstitial request sent",
                           Toast.LENGTH_SHORT).show();

            break;
        default:
            }

    }

    @Override
    public void onInterstitialCompleted() {
        Toast.makeText(MainActivity.this, "Interstitial play completed",
                       Toast.LENGTH_SHORT).show();
    }

    @Override
    public void onInterstitialStarted() {
        Toast.makeText(MainActivity.this, "Interstitial started",
                       Toast.LENGTH_SHORT).show();
    }
}

```

## Showing Pre Roll Ads

Wherever you want to show an Pre Roll Video ad, declare a BFIOPreRoll instance variable and instantiate the instance in the onCreate(Bundle savedInstanceState) method, as shown below:

```

public class MainActivity extends Activity {

    BFIOPreRoll preRoll;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        preRoll = new BFIOPreRoll(this);
    }
}

```

If you want to see the pre roll call back events. You can instantiate in a different way:

```
public class MainActivity extends Activity implements  
PreRollAdListener {  
  
    BFIOPreRoll preRoll;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        preRoll = new BFIOPreRoll(this, this);  
    }  
}
```

You need to use IO Video View to show pre roll video ads. IO Video View in an android custom view & can be declare in layout xml files :

```
<com.bfio.ad.VideoViewIO  
    android:id="@+id/io_video_view"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent" />
```

IO Video View can also build programmatically using java code:

```
VideoViewIO videoViewIO = new VideoViewIO(context);
```

Your application is now ready to display an Pre Roll ad. Now when you are ready to play a video with pre roll, call the playWithPreRoll method, passing in your appld, adUnitId & IOVideoView:

```
// Set the url of actual video  
ioVideoView.setVideoURI(uri);  
preRoll.playWithPreRoll("<appId>", "<adUnitID>", ioVideoView);
```

In case of successful fill, it will play the pre roll first then the actual video, otherwise will play the actual video

Following are callbacks methods of an pre roll ad:

```
/**
 * PreRoll did not found
 *
 * @param errorCode
 */
public void onFailed(BFIOErrorCode errorCode);

/**
 * PreRoll Ad started
 *
 */
public void onPreRollStarted();

/**
 * PreRoll clicked
 *
 */
public void onPreRollClicked();

/**
 * PreRoll completed
 *
 */
public void onPreRollCompleted();
```