

Show Native Ads on Android

Show Native Ads on Android

Native ads give you the ability to create ads that are customized to match the look and feel of the rest of your application. This page describes our Native Ads API at a high level, and includes a usage example. For a complete reference, see the [Javadoc](#).

Native networks supported through **mediation**:

- Mopub
- inMobi
- Facebook
- Yahoo Flurry
- AdMob and DFP

In order to serve native ads, you will send a native ad request, and receive a native ad response. For Android 9 and above and API v. 28 and above, the request must be HTTPS by default in order to track viewability accurately. You can enable HTTPS with `useHttps(true)`.

In the example code below, we:

- Set up a request object and supply it with either:
 - The placement ID (as shown in the example code below), OR
 - A combination of inventory code and member ID:

```
public NativeAdRequest nativeAdRequest= new NativeAdRequest(context,
    "PLACEMENT_ID" );
//public NativeAdRequest nativeAdRequest= new NativeAdRequest(context,
    "INVENTORY_CODE", MEMBER_ID);
```

- Optionally, you can set the `renderer_id` for this `NativeAdRequest`. (For more on `renderer_id` see [Native Layout Service](#).) The `renderer_id` needs to be specified in order for `vastxml`, `likes`, `downloads`, `saleprice`, `phone`, `address`, and `display URL` to be returned in the `NativeAdResponse`.

```
nativeAdRequest.setRendererId(RENDERER_ID);
```

- Register a listener that will signal native ad events such as clicks (`NativeAdEventListener`).
- Register a listener to signal the state of the native request: success or failure. The listener must implement the `NativeAdRequestListener` interface.
- If the request is successful (i.e., `NativeAdListener.onAdLoaded()` fires), native ad assets are loaded in the `NativeAdResponse` object which can be used in views that match the native look of the app. Then register the parent or container view of these views to enable impression and click tracking.
- Unregister the native ad view after we're finished with it.

Maintain references to native views and native response objects

Maintain references to native views and native response objects.

It is your responsibility to keep a reference to the native ad view and `NativeAdResponse` object if necessary.

```
public class MyActivity extends Activity {

    Context activityContext;
    NativeAdResponse nativeAdResponse;
    LinearLayout container;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        activityContext = this;
```

```

        // Create a NativeAdRequest object
        NativeAdRequest adRequest = new NativeAdRequest(activityContext, "123456"); //
Placement ID

        // Optionally set the renderer_id
        //adRequest.setRendererId(123);

        // Create a listener for ad events
        NativeAdEventListener adEventListener = new
            NativeAdEventListener() {
                @Override
                public void onAdWasClicked() {
                    // Do something when the view is clicked
                }

                @Override
                public void onAdWillLeaveApplication() {
                    // Do something when the ad is taking user away from current
app
                }

                @Override
                public void onAdWasClicked(String clickUrl, String fallbackURL) {
                    // Handle Click URL
                }
            };

        // Whether to pre-load the native ad's icon and main image
        adRequest.shouldLoadIcon(true);
        adRequest.shouldLoadImage(true);

        adRequest.setListener(new NativeAdRequestListener() {
            @Override
            public void onAdLoaded(NativeAdResponse response) {
                nativeAdResponse = response;
                // Cover image
                ImageView imageView = new ImageView(activityContext);
                imageView.setImageBitmap(response.getImage());

                // Icon image
                ImageView iconView = new ImageView(activityContext);
                iconView.setImageBitmap(response.getIcon());

                // Title
                TextView title = new TextView(activityContext);
                title.setText(response.getTitle());

                // Main text
                TextView description = new TextView(activityContext);
                description.setText(response.getDescription());

                // Text that indicates a call to action -- for example, to install an
app
                TextView callToAction = new TextView(activityContext);
                callToAction.setText(response.getCallToAction());

                // Create a container (a parent view that holds all the
                // views for native ads)

```

```
        LinearLayout container = new LinearLayout(activityContext);
        container.addView(iconView);
        container.addView(title);

        // Add the native ad container to the view hierarchy
        LinearLayout ad_frame = findViewById(R.id.native_ad_frame);
        ad_frame.addView(container);
    }

    @Override
    public void onAdFailed(ResultCode errorcode) {

    }
});

// Call loadAd() to request a response once
adRequest.loadAd();

// Register native views for click and impression tracking. The
// adEventListener is the listener created above; it can be null if
// you don't want to receive notifications about click events.
// Impressions and clicks won't be counted if the view is not registered.
NativeAdSDK.registerTracking(nativeAdResponse, container, adEventListener);

// It's your responsibility to keep a reference to the view
// and NativeAdResponse object if necessary.
// Once done with the native ad view, call the following method to
// unregister that view.
NativeAdSDK.unregisterTracking(container);
```

```
}
}
```

Fields Supported in Native

As of version 5.0 of the Mobile SDK, support for native assets is aligned with how native creatives are set up on Console.

If you are still using Legacy Native in Console, you will need to move to "New" Native for your creatives.

The following is a comprehensive list of native assets supported in the SDKs.

Asset	Supported Pre 5.0?	Supported Post 5.0?	v5.0+ API-Usage Example	
Image, Width, Height	Yes, Yes, Yes	Yes, Yes, Yes	<pre>nativeAdResponse.getImage(); nativeAdResponse.getImageSize(); nativeAdResponse.getImageUrl();</pre>	
Icon+Width+Height	Yes, No, No	Yes, Yes, Yes	<pre>nativeAdResponse.getIcon(); nativeAdResponse.getIconSize(); nativeAdResponse.getIconUrl();</pre>	
Title	Yes	Yes	<pre>nativeAdResponse.getTitle();</pre>	
Sponsored by	Yes	Yes	<pre>nativeAdResponse.getSponsoredBy();</pre>	
Body text	Yes	Yes	<pre>nativeAdResponse.getDescription();</pre>	
Desc2	Yes	Yes	<pre>nativeAdResponse.getAdditionalDescription();</pre>	
Call-to-action	Yes	Yes	<pre>nativeAdResponse.getCallToAction();</pre>	
Rating, Scale	Yes, Yes	Yes, No	<pre>nativeAdResponse.getAdStarRating();</pre>	
Likes	No	Yes (json only)	<pre>if ((nativeAdResponse.getNetworkIdentifier() == NativeAdResponse.Network.APPNEXUS) (nativeAdResponse.getNativeElements().get(NativeAdResponse.NATIVE_ELEMENT_OBJECT)) JSONObject nativeResponseJSON = (JSONObject) (nativeAdResponse.getNativeElements().get(NativeAdResponse.NATIVE_ELEMENT_OBJECT)) String likes = JsonUtil.getJSONString(nativeResponseJSON, "likes"); String downloads = JsonUtil.getJSONString(nativeResponseJSON, "downloads"); String price = JsonUtil.getJSONString(nativeResponseJSON, "price"); String saleprice = JsonUtil.getJSONString(nativeResponseJSON, "saleprice"); String phone = JsonUtil.getJSONString(nativeResponseJSON, "phone"); String address = JsonUtil.getJSONString(nativeResponseJSON, "address"); String displayurl = JsonUtil.getJSONString(nativeResponseJSON, "displayurl"); }</pre>	
Downloads	No	Yes (json only)		
Price	No	Yes (json only)		
Sale Price	No	Yes (json only)		
Phone	No	Yes (json only)		
Address	No	Yes (json only)		
Display URL	No	Yes (json only)		
Privacy URL	No	Yes		<pre>nativeAdResponse.getPrivacyLink();</pre>
Video	No	Yes		<pre>nativeAdResponse.getVastXml();</pre>
Custom	Yes	No		
Context	Yes	No		

Full text	Yes	No	
-----------	-----	----	--

Related Topics

- [Integrate with Android](#)
- [Mediate with Android](#)