

```
1 package com.catdog2025.mediation.java;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import androidx.annotation.Nullable;
6 import android.util.Log;
7 import android.view.View;
8 import android.widget.FrameLayout;
9 import android.widget.TextView;
10
11 import com.bytedance.sdk.openadsdk.AdSlot;
12 import com.bytedance.sdk.openadsdk.TTAdDislike;
13 import com.bytedance.sdk.openadsdk.TTAdNative;
14 import com.bytedance.sdk.openadsdk.TTNativeExpressAd;
15 import com.catdog2025.R;
16 import com.catdog2025.config.TTAdManagerHolder;
17 import com.catdog2025.mediation.java.utils.Const;
18 import com.catdog2025.utils.UIUtils;
19
20 import java.util.List;
21
22 /**
23  * 融合demo，banner广告使用示例。更多功能参考接入文档。
24  *
25  * 注意：每次加载的广告，只能展示一次
26  *
27  * 接入步骤：
28  * 1、创建AdSlot对象
29  * 2、创建TTAdNative对象
30  * 3、创建加载、展示监听器
31  * 4、加载广告
32  * 5、加载成功后，展示广告
33  * 6、在onDestroy中销毁广告
34  */
35 public class MediationBannerActivity extends Activity {
36
37     private static final String TAG = "MediationBannerActivity";
38
39     public String mMediaId; // 融合广告位
```

```

40
41     private TTNativeExpressAd mBannerAd; // Banner广告对象
42
43     private TAdNative.NativeExpressAdListener mBannerListener; //
        广告加载监听器
44
45     private TTNativeExpressAd.ExpressAdInteractionListener
        mBannerInteractionListener; // 广告展示监听器
46
47     private TAdDislike.DislikeInteractionCallback mDislikeCallback;
        // 接受广告关闭回调
48
49     private FrameLayout mBannerContainer; // banner广告容器view
50
51     @Override
52     protected void onCreate(@Nullable Bundle savedInstanceState) {
53         super.onCreate(savedInstanceState);
54         setContentView(R.layout.mediation_activity_banner);
55
56         // 聚合广告位 ( 在GroMore平台的广告位，注意不是adn的代码位
        )
57         this.mMediaId = getResources().getString(R.string.
        banner_media_id);
58         TextView tvMediationId = (TextView)this.findViewById(R.id.
        tv_media_id);
59         tvMediationId.setText(getString(R.string.ad_mediation_id, this.
        mMediaId));
60
61         // banner广告容器
62         mBannerContainer = findViewById(R.id.banner_container);
63
64         // 广告加载
65         findViewById(R.id.bt_load).setOnClickListener(new View.
        OnClickListener() {
66             @Override
67             public void onClick(View v) {
68                 loadBannerAd();
69             }
70         });

```

```

71
72     // 广告展示
73     findViewById(R.id.bt_show).setOnClickListener(new View.
OnClickListener() {
74         @Override
75         public void onClick(View v) {
76             showBannerAd();
77         }
78     });
79 }
80
81 private void loadBannerAd() {
82     /** 1、创建AdSlot对象 */
83
84     AdSlot adSlot = new AdSlot.Builder()
85         .setCodeId(this.mMediaId)
86         .setImageAcceptedSize(UIUtils.dp2px(this, 320f), UIUtils.
dp2px(this, 150f)) // 单位px
87         .build();
88
89     /** 2、创建TTAdNative对象 */
90
91     TTAdNative adNativeLoader = TTAdManagerHolder.get().
createAdNative(this);
92
93     /** 3、创建加载、展示监听器 */
94     initListeners();
95
96     /** 4、加载广告 */
97     if (adNativeLoader != null) {
98
99         adNativeLoader.loadBannerExpressAd(adSlot,
mBannerListener);
100     }
101 }
102
103 /** 5、广告加载成功后，设置监听器，展示广告 */
104 private void showBannerAd() {
105     if (mBannerAd != null) {

```

```

106         mBannerAd.setExpressInteractionListener(
            mBannerInteractionListener);
107         mBannerAd.setDislikeCallback(this, mDislikeCallback);
108         mBannerAd.uploadDislikeEvent("mediation_dislike_event"
            );
109         /** 注意：使用融合功能时，
            load成功后可直接调用getExpressAdView获取广告view展示，
            而无需调用render等onRenderSuccess后 */
110
111         View bannerView = mBannerAd.getExpressAdView();
112         if (bannerView != null && mBannerContainer != null) {
113             mBannerContainer.removeAllViews();
114             mBannerContainer.addView(bannerView);
115         }
116     } else {
117         Log.d(Const.TAG, "
            请先加载广告或等待广告加载完毕后再展示广告");
118     }
119 }
120
121 private void initListeners() {
122     // 广告加载监听器
123
124     this.mBannerListener = new TTAdbNative.
        NativeExpressAdListener() {
125         @Override
126         public void onError(int i, String s) {
127             Log.d(Const.TAG, "banner load fail: errCode: " + i + ",
            errMsg: " + s);
128         }
129
130         @Override
131
132         public void onNativeExpressAdLoad(List<
            TTNativeExpressAd> list) {
133             if (list != null && list.size() > 0) {
134                 Log.d(Const.TAG, "banner load success");
135                 mBannerAd = list.get(0);
136             } else {

```

```

137         Log.d(Const.TAG, "banner load success, but list is null"
138     );
139     }
140 };
141 // 广告展示监听器
142
143     this.mBannerInteractionListener = new TTNativeExpressAd.
ExpressAdInteractionListener() {
144         @Override
145
146         public void onAdClicked(View view, int i) {
147             Log.d(Const.TAG, "banner clicked");
148         }
149
150         @Override
151
152         public void onAdShow(View view, int i) {
153             Log.d(Const.TAG, "banner showed");
154         }
155
156         @Override
157         public void onRenderFail(View view, String s, int i) {
158             // 注意：使用融合功能时，无需调用render，
load成功后可调用mBannerAd.getExpressAdView()进行展示。
159         }
160
161         @Override
162         public void onRenderSuccess(View view, float v, float v1) {
163             // 注意：使用融合功能时，无需调用render，
load成功后可调用mBannerAd.getExpressAdView()
获取view进行展示。
164             // 如果调用了render，则会直接回调onRenderSuccess
，***** 参数view为null，请勿使用。*****
165         }
166     };
167     // dislike监听器，广告关闭时会回调onSelected
168
169     this.mDislikeCallback = new TTAdDislike.

```

```
169 DislikeInteractionCallback() {
170     @Override
171     public void onShow() {}
172
173     @Override
174     public void onSelected(int i, String s, boolean b) {
175         if (mBannerContainer != null)
176             mBannerContainer.removeAllViews();
177         Log.d(Const.TAG, "banner closed");
178     }
179
180     @Override
181     public void onCancel() {}
182 };
183 }
184
185 @Override
186 protected void onDestroy() {
187     super.onDestroy();
188     /** 6、在onDestroy中销毁广告 */
189     if (mBannerAd != null) {
190         mBannerAd.destroy();
191     }
192 }
193 }
194
195
```