

APP广告分发平台SDK接入流程

1. 准备

1. 获取广告平台开发者账号和APP的唯一标识（16个字符的App ID），区分Android和iOS；
2. 登录App广告管理后台系统，进入 APP广告管理->应用管理，便可以看到上一步申请到的应用信息，然后进入 APP广告管理->广告位管理，新建对应的广告位，获取广告位ID。

2. Android APP接入SDK

2.1 导入SDK依赖包

```
compile 'com.github.bumptech.glide:glide:3.5.2'  
compile 'com.squareup.okhttp3:okhttp:3.9.0'  
compile 'com.google.code.gson:gson:2.8.1'
```

2.2 开屏广告

首先添加用来展示开屏广告的view

```
<com.phicomm.adplatform.startPage.StartPage  
    android:id="@+id/startPage"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:scaleType="fitXY" />
```

代码中使用SDK：

```
StartPage startPage = (StartPage) findViewById(R.id.startPage);  
startPage.setOnStartPageClickListener(this);  
startPage.start(appId, placementId); //传入应用ID，广告位ID
```

实现 `OnStartPageListener` 接口，可以实现对点击开屏图片事件和跳过图片事件的监听。

```

@Override
public void OnImageClick() {
    // 点击开屏图片
    //获取点击的开屏图片落地页和标题
    startPage.getLandPage();
    startPage.getTitle();
}

@Override
public void OnSkipClick() {
    // 点击跳过按钮
}

@Override
public void onADDismissed() {
    // 广告加载完成
}

@Override
public void onNoAD() {
    // 广告加载失败
}

```

2.3 横幅广告

首先添加用来展示横幅广告的view

```

<com.phicomm.adplatform.banner.Banner
    android:id="@+id/banner"
    android:layout_width="match_parent"
    android:layout_height="200dp" />

```

其中 `layout_width` 和 `layout_height` 属性的值要根据在管理后台中新建的横幅广告位的宽和高来设置。

代码中使用SDK:

```

Banner banner = (Banner) findViewById(R.id.banner);
banner.setOnBannerListener(this);
banner.start(appId, placementId); //传入应用ID和广告位ID

```

实现 `OnBannerListener` 接口，可以实现对点击横幅图片事件的监听。

获取广告点击时跳转的链接和标题

```

@Override
public void OnBannerClick(int position) {
    //获取点击的Banner图片落地页和标题
    banner.getLandPage(position);
    banner.getTitle(position);
}

```

3. IOS APP接入SDK

3.1 IOS SDK依赖包

```

FLAnimatedImageView
SDWebImage
AFNetworking
JSONModel

```

3.2 开屏广告

1. 在需要使用的ViewController中导入: `#import "ADManager.h"`;
2. 需要实现协议ADSetupActionDelegate, 这样这个ViewController就可以监听启动广告页面的一些事件

```

/**
 * 跳过按钮被点击
 */
- (void)skipButtonClickedWithController:(UIViewController *) controller;

/**
 * 开屏广告被点击
 */
- (void)imageClickedWithController:(UIViewController *) controller title:(NSString *)title landPage:(NSString *)landPage;//获取落地页和标题

/**
 * 广告展示结束回调
 */
- (void)adShowFinishedWithController:(UIViewController *) controller;

/**
 * 没有广告回调
 */
- (void)noAdShow;

```

示例如下:

```
[self performSelector:@selector(go) withObject:self afterDelay:1.0f];
- (void)go{
    [ADManager initWithController:self appId:@"apzrnbi0fzpe67bn" placement
    Id:21];//第一个参数是当前的ViewController，第二个参数和第三个参数传入应用ID和广告位
    ID。
}
```

3.3 横幅广告

在对应的ViewController中 导入： `#import "UIBannerView.h"`，实现协议 `ADBannerImageDelegate`

```
/**
 * 横幅广告被点击
 */
- (void)adBannerImageClicked:(NSString *)title landPage:(NSString *)landPage; //获取落地页和标题
```

示例如下：

```
UIBannerView *bannerView = [[UIBannerView alloc] init];
[bannerView workWithAppId:appId placementId:placementId]; //传入应用ID和广告位ID
bannerView.frame = CGRectMake(0, 0, self.view.frame.size.width, 200); //根据广告后台配置的高度和宽度设置
[self.view addSubview:bannerView];
```

4. 广告加载时使用的策略

开屏：

1. 先请求广告后台服务器获取是否应该展示广告；
2. 如果可以展示则加载上次请求后台的缓存，如果不可以则不展示；
3. 访问服务器超时时间为1S，超时则加载缓存；
4. 每次请求服务器会缓存最新的广告，用于下次展示。

横幅：

1. 先请求广告后台服务器获取是否应该展示广告；
2. 如果可以展示则展示本次请求到的结果，如果不可以则不展示；
3. 访问服务器超时时间为2s，超时则加载缓存；
4. 每次请求服务器会缓存最新的广告。

5. 注意事项

- 获取到落地页后需要判断落地页是否为空字符串或Null来决定是否跳转。
- 如APP使用内置浏览器需在webview中设置独立UA标识，并告知平台联系人

6. 平台测试环境地址

<https://phicloudsymtest.phicomm.com/ADPush>

7. 平台开发联系人

- SDK：卞一夫
- 后台：韩仁彬

