

Avazu Native Ad SDK Setup for iOS

Date	Version	Description	Developer
2015.03.12	v1.0.0	First Version	曹诗聪
2015.03.25	v1.0.1	Optimization and issue fix	曹诗聪

1. 简介	3
1.1 SDK功能介绍	3
1.2 兼容性	3
1.3 开发环境要求	3
2. 接入配置	4
2.1 Traffic Source ID	4
2.2 导入SDK文件	4
2.3 添加依赖库	4
3. 展示广告	5
3.1 创建AvazuADView实例	5
3.2 选择一个适合您的广告形式	6
3.3 定制化广告	6
4. 实现delegate	8

1.简介

1.1 SDK功能介绍

Avazu ADSDK 能够提供以下功能

- 展示单个广告banner (推荐尺寸: 320 * 100)
- 展示透明banner (推荐尺寸: 400 * 110)
- 展示广告banner墙
- 展示单行矩形广告墙
- 展示多行矩形广告墙

以上这些广告展现方式, 开发者可以根据SDK提供的方法实现广告外观和展示应用数量的定制

1.2 兼容性

支持 iOS6.0 以上的系统

1.3 开发环境要求

操作系统: macosx lion及以上版本

开发工具: Xcode 5.0及以上版本

2. 接入配置

2.1 Traffic Source ID

确保您已注册Avazu APX账号且已有可用的Traffic Source ID for Display

2.2 导入SDK文件

请将以下文件添加到您的项目中：

- [libAvazuAdSDK.a](#)
- [AvazuADView.h](#)






2.3 添加依赖库

为了保证SDK能够正确运行，您需要添加以下依赖库到项目中：

- [CoreTelephony.framework](#)
- [UIKit.framework](#)
- [Foundation.framework](#)
- [AdSupport.framework](#)

▼ Link Binary With Libraries (5 items)

×

Name	Status
 CoreTelephony.framework	Required ⇅
 libAvazuAdSDK.a	Required ⇅
 UIKit.framework	Required ⇅
 Foundation.framework	Required ⇅
 AdSupport.framework	Required ⇅
+ — Drag to reorder frameworks	

3. 展示广告

AvazuADSDK 提供了5种不同种类的广告展现形式，包括单个广告banner，广告banner墙，单行矩形广告墙，多行矩形广告墙，透明banner。

作为开发者，您只需要通过**AvazuADView**这个接口来创建这些广告类型，**AvazuADView**是UIView的子类，可以展现HTML5的广告。

您可以遵从以下步骤来成功展示广告：

3.1 创建**AvazuADView**实例

您需要确认您的 Traffic Source ID是可用的，每次创建广告实例时都需要用到它，否则将无法正确展示广告。

创建步骤如下：

1. Import **AvazuADView.h** 文件.
- 2.在头文件中声明一个实例，您的头文件类似如下所示：

```
#import <UIKit/UIKit.h>
#import "AvazuADView.h"

@interface ViewController : UIViewController <AvazuADViewDelegate>
@property (strong, nonatomic) AvazuADView *adView;
@end
```

3. 在实现文件中创建实例，范例如下：

```
self.adView = [[AvazuADView alloc] initWithFrame:adFrame
                                           adType:AVAZU_SINGLE_BANNER
                                           sourceID:@"input your Traffic Source ID here"];
```

4. 创建实例后，在载入广告之前您需要在设置该实例的delegate, 用来通知应用广告载入成功或失败。
5. 将**adView**添加到主view中
6. 根据您需求定制广告的展现形式和展示应用数量，我们将会3.3章节中做详细说明
7. 用此方法来载入广告：**[self.adView loadAD]**

以下是Viewcontroller中的一个完整范例：

```

- (void)viewDidLoad {
    [super viewDidLoad];
    CGRect adframe = CGRectMake(0.0, 0.0, 320.0, 100.0);
    self.adView = [[AvazuADView alloc] initWithFrame:adframe
                                              adType:AVAZU_SINGLE_BANNER
                                              sourceID:@"1234"];

    self.adView.delegate = self;
    [self.view addSubview:self.adView];
    //Insert Customized Advview setup here
    [self.adView loadAD];
}
    
```

8. 在**ViewController**中, 我们推荐您在dealloc函数将delegate设为nil, 或在任意释放adView的地方您也需要将其设为nil

```

- (void)dealloc {
    self.adView.delegate = nil;
}
    
```

3.2 选择一个适合您的广告形式

广告形式	值	变量名
单个广告banner	9	AVAZU_SINGLE_BANNER
banner广告墙	10	AVAZU_BANNER_APPWALL
透明banner	11	AVAZU_TRANSPARENT_SINGLE_BANNER
单行矩形广告墙	12	AVAZU_SINGLE_BUTTON_APP_WALL
多行矩形广告墙	13	AVAZU_MULTIPLE_BUTTON_APP_WALL

3.3 定制化广告

AvazuADSDK支持丰富的定制化广告, 您可以选择展示的广告的颜色搭配, 展示内容以及展示应用的数量, 只需要像如下的代码进行设置就可:

```

self.adView.isNeedIcon = 0;    //不显示广告中的应用图片
    
```

1. 广告元素配置

变量名	类型	默认值	作用
isNeedIcon	BOOL	1	设置广告中是否显示应用图标
isNeedTitle	BOOL	1	设置广告中是否显示应用名称
isNeedCat	BOOL	1	设置广告中是否显示应用类别
isNeedSize	BOOL	1	设置广告中是否显示应用尺寸
isNeedRating	BOOL	1	设置广告中是否显示应用评分
isNeedInstallButton	BOOL	1	设置广告中是否显示安装按钮
isNeedReviewNumber	BOOL	1	设置广告中是否显示应用查看次数
isNeedLoadingIndicator	BOOL	1	设置加载广告时是否显示载入时圆圈

2. 广告颜色配置

注意: 配置广告颜色时, 颜色所有颜色值的设置方式为“#”加上6位十六进制数RGB格式的字符串, 如“#FFFFFF”表示白色, “#000000”表示黑色

变量名	类型	默认值	作用
blockBackColor	NSString	null	设置广告块背景色
appTitleColor	NSString	null	设置应用名字体颜色
buttonBackColor	NSString	null	设置下载按钮颜色
buttonTextColor	NSString	null	设置下载按钮上的字体颜色
mainBackColor	NSString	null	设置应用墙背景色

3. 展示应用数量配置

变量名	类型	默认值	作用
appCount	INT	1	设置广告中出现的应用数量

注意: 以下广告类型必须设置appCount:

广告banner墙, 单行矩形广告墙, 多行矩形广告墙

4. 设置广告透明度

变量名	类型	默认值	作用
transparentBannerAlpha	INT	0	设置透明广告的透明度

注意: 只有在透明banner的广告形式下才用设置transparentBannerAlpha, 合法取值范围0~100,0表示完全透明, 100表示完全不透明

以下是一个定制广告banner墙的完整代码:

```
- (void)viewDidLoad {
    [super viewDidLoad];
    CGRect adframe = CGRectMake(0.0, 0.0, 320.0, 100.0);
    self.adView = [[AvazuADView alloc] initWithFrame:adframe
                                           adType:AVAZU_BANNER_APPWALL
                                           sourceID:@"1234"];

    self.adView.delegate = self;
    [self.view addSubview:self.adView];
    self.adView.isNeedSize = 0;
    self.adView.isNeedReviewNumber = 0;
    self.adView.appCount = 4;
    [self.adView loadAD];
}
```

4. 实现delegate

如果您需要知道广告显示结果的回调, 请在应用中实现如下两个delegate方法, 在delegate中您可以知道这次广告是否展示成功, 您可以根据应用所需逻辑重写该方法。

```
- (void)avazuADViewLoadAdSucess:(AvazuADView *)adview
{
    NSLog(@"avazuADViewLoadAdSucess");
}

- (void)avazuADView:(AvazuADView *)adview didFailToReceiveAdWithError:(NSError *)error
{
    NSLog(@"avazuADViewLoadAdFail");
    NSLog(@"error:%@", error);
}
```