

iOS SDK

How to Use this SDK in Your IOS Applications

Step 1. Get the code

Download Mobile SDK files from your zone near to Invocation code

Step 2. Extracting the SDK Files

Extract that .zip file then copy the library file into your application folder

Step 3. Add library files into your application

Library file named libdJAX_Lib.a is available in the zip file and include folder for header file, Add this library files into your application and use this library file by following the code structure given below,

Sample code:

- To import the library file

```
#import "dJAX_Lib.h"
```

- Basic, MRAID and IAB ad formats zone_id's are passed using the below code,

```
[[[Ad_Response alloc] init] PassZoneid:28]; //Pass zoneid here
```

PassZoneid:28 - You can enter your zone id for any ad formats; replace 28 with your zone id

BAISC / MRAID / IAB Ad Formats:

- Get Basic, MRAID, IAB ad formats value by using the below code,

```
NSString *ad_value = [[[Ad_Response alloc] init] ad];
```

• To Display the Basic, MRAID, IAB ad formats the generated ad_value needs to be passed inside the webview as loadHTMLString

```
[self.Webview loadHTMLString:ad_value baseURL:nil];
```

Ad Display Transparency Control:

- For Ad transparency you can add these code to display ad without **white background**

```
[_Webview setBackgroundColor:[UIColor clearColor]];  
[_Webview setOpaque:NO];
```

Auto Rotate Ad:

- Auto Rotate ad with the NSTimer, you can refresh the selector

```
[NSTimer scheduledTimerWithTimeInterval:20 target:self selector:@selector(viewDidLoad)
userInfo:nil repeats:false];
```

scheduledTimerWithTimeInterval – Auto rotate time interval will be included in this, Here instead 20 you can enter your time interval in seconds

selector:@selector(viewDidLoad) – Selector value passed instead of viewDidLoad

After Integration:

Integrated application will display the specified ad as shown below,

**NOTE:**

- Change your build settings architecture to \$(ARCHS_STANDARD_32_BIT)
- If you are using iOS9, include the below code in **info.plist** file (For security reason MAC OS X EI Capitan has blocked the https)

```
<key>NSAppTransportSecurity</key>
<dict> <key>NSAllowsArbitraryLoads</key><true/>
</dict>
```