

#SuperG SDK for iOS

Revision history

3.0.2	<ul style="list-style-type: none">-Removed ad serving: Just add your own ad serving SDK. Cliq Digital works with MADS. You can download the MADS SDK here: http://developer.madsone.com/IOS_SDK-Removed the Thumbr subscription flow-Removed all assets that are no longer needed-Added Generic Event-Removed mandatory settings. All is automatically set within the SDK now.
3.0.1	<ul style="list-style-type: none">-Bugfixes
3.0.0	<ul style="list-style-type: none">-Rebranded to SuperG-Event logging-Push Notifications Support-Separate Thumbr T animation Framework-AppsFlyer made optional-iOS 7 Support-Removal of deprecated methods-Performance optimizations
2.0.323	<ul style="list-style-type: none">-Removed status bar after stopAds-Added customizable CGPoint for inline ads
2.0.322	<ul style="list-style-type: none">-Added stopAds function (currently loading ads will be cancelled)
2.0.321	<ul style="list-style-type: none">-Ad overlay bug fix
2.0.32	<ul style="list-style-type: none">-Possibility to hide the Thumbr close button (remote configuration)-Improved interstitial ads-Added demo and manual for Ad-only integration
2.0.31	<ul style="list-style-type: none">-Personalized Ad serving support-Technical updates
2.0.3	<ul style="list-style-type: none">-Ad serving support-SSL support-Technical updates
2.0.22	<ul style="list-style-type: none">-Fixed memory leaks, improved memory clean up-Removed refresh button in iPad layout-Catch url's that contain 'openinbrowser' and open them in browser-Send along SDK version number to Thumbr server-Improved orientation behavior-Upgraded to Xcode 4.6.1 requirements
2.0.21	<ul style="list-style-type: none">-Better Thumbr T-button resizing-Added visual SDK version to Thumbr window-Added Customizable Orientation-Added extra parameters to registration flow: default,registration,optional_registration-Added a counter to the 'opens' of the Thumbr SDK window-Let external URL's (within the SDK window) be opened in the default browser-Bug fix: better orientation handling in general-Let Thumbr server know that back end is loaded from within the SDK (via GET param (&sdk=1) + via HEADER (x-thumbr-method))-Added version header (X-Thumbr-Version)
2.0.2	<ul style="list-style-type: none">-Updated version number to match iOS version (for better release planning)-Added Animated Thumbr T-button support-Added SDK version number to Thumbr screen-Bug fixes
1.1	<ul style="list-style-type: none">- Added Appsflyer support
1.0.1	<ul style="list-style-type: none">- Bug fixes

Easy SDK Setup

Step 1

Clone this repository to you local disk and drag the ****SDK Files**** into the file browser of your XCode project (Copy files and create group!)

Add this to the 'Other linker flags' in (your project)->Build Settings->Linking:

`-all_load -ObjC`

Make sure that the following frameworks are included in your Build Phases->Link Binaries With Libraries:

Name	Status
Internal frameworks	
AdSupport.framework	Required
AudioToolbox.framework	Required
AVFoundation.framework	Required
CFNetwork.framework	Required
CoreAudio.framework	Required
CoreLocation.framework	Required
CoreTelephony	Required
EventKit.framework	Required
Foundation.framework	Required
libz.dylib	Required
libsqlite3.0.dylib	Required
MediaPlayer.framework	Required
MessageUI.framework	Required
SystemConfiguration.framework	Required
UIKit.framework	Required
QuartzCore.framework	Required
External libraries (from the SDK package)	
SuperG.framework	Required
LRResty.framework	Required

Step 2

Add the .caf files to your project root. These are the custom push notification sounds.

Step 3

Open your AppDelegate.h file and add to the top:

```
#import "SuperG/SuperG.h"
#import "SuperG/PUSH.h"
#import "SuperG/EVA.h"

@class ViewController; //<=The name of your main view controller
```

Then declare (or add to existing delegate):

```
@interface AppDelegate : UIResponder <UIApplicationDelegate, SuperGSDKDelegate>
```

Step 4

In your appDelegate.m, add (or update) these methods:

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {
    //..other code...

    [SuperG initializeSDK:self];
    return YES;
}

- (void)applicationWillEnterForeground:(UIApplication *)application {
    [SuperG initializeSDK:self];
}

- (void)applicationWillTerminate:(UIApplication *)application {
    [SuperG stop];
}

- (void)applicationDidEnterBackground:(UIApplication *)application {
    [SuperG stop];
}

- (void)application:(UIApplication *)application didReceiveRemoteNotification:(NSDictionary *)userInfo {
    [PUSH handle:userInfo :[application applicationState]];

    //HANDLE YOUR OWN APPLICATION SPECIFIC ACTIONS AFTER PUSH NOTIFICATION HERE:
    //NSString *message = [[userInfo valueForKey:@"aps"] valueForKey:@"alert"];
    //NSString *action = [[userInfo valueForKey:@"aps"] valueForKey:@"action"];
}
```

Step 5 :: Events

If you want to log custom events, add this line to your imports:

```
#import "SuperG/EVA.h"
```

These custom event methods are available:

```
+ (void) click:(NSString*)clicked_item;
    Example:
[EVA click:@"TestButton"];

+(void)purchase: (NSString*)purchased_item WithCurrency: (NSString*)currency PaymentMethod:(NSString*)payment_method AndPrice:(NSString*)price
    Example:
[EVA purchase:@"coins" WithCurrency:@"EUR" PaymentMethod:@"in-app" AndPrice:@"0.79"];

+ (void) achievementEarned:(NSString*)achievement_name
    Example:
[EVA achievementEarned:@"Found gold"];

+ (void) start_level:(NSString*)level inMode:(NSString*)game_mode withScoreType: (NSString*)score_type andScoreValue:(NSString*)score_value
    Example:
[EVA start_level:@"1" inMode:@"attach_mode" withScoreType:@"win" andScoreValue:@"1"];

+ (void) finish_level:(NSString*)level inMode:(NSString*)game_mode withScoreType: (NSString*)score_type andScoreValue:(NSString*)score_value
    Example:
[EVA finish_level:@"100" inMode:@"attack mode" withScoreType:@"win" andScoreValue:@"1" ];

+(void)upSell:(NSString*)currency PaymentMethod:(NSString*)payment_method
    Example:
[EVA upSell:@"EUR" PaymentMethod:@"in-app"];

+(void)generic: (NSString*)generic_key GenericValue: (NSString*)generic_value
    Example:
[EVA generic:@"myEvent" GenericValue:@"it fired!"];
```