Spil iOS Framework

Generated by Doxygen 1.8.1.2

Thu Oct 10 2013 14:13:51

Contents

1 Deprecated List 1

2 Class Index 3

2.1 Class List . . . 3

3 Class Documentation 5

3.1 ABTestAdapter Class Reference . . . 5

3.2 <ABTestDelegate> Protocol Reference . . . 5

3.2.1 Detailed Description . . . 5

3.2.2 Member Function Documentation . . . 6

3.2.2.1 abtestSessionDidEnd . . . 6

3.2.2.2 abtestSessionDidStart . . . 6

3.2.2.3 abtestSessionDiffReceived: . . . 6

3.3 <AdsDelegate> Protocol Reference . . . 6

3.3.1 Detailed Description . . . 6

3.3.2 Member Function Documentation . . . 7

3.3.2.1 adDidAppear . . . 7

3.3.2.2 adDidFailToAppear: . . . 7

3.3.2.3 adDidFailToStart: . . . 7

3.3.2.4 adDidStart . . . 7

3.3.2.5 adMoreGamesDidAppear . . . 7

3.3.2.6 adMoreGamesDidDismiss . . . 7

3.3.2.7 adMoreGamesDidFailToAppear: . . . 7

3.3.2.8 adMoreGamesWillAppear . . . 7

3.3.2.9 adPopupDidDismiss . . . 8

3.3.2.10 adWillAppear . . . 8

3.4 <AppSettingsDelegate> Protocol Reference . . . 8

3.4.1 Detailed Description . . . 8

3.4.2 Member Function Documentation . . . 8

3.4.2.1 appSettingsDidFailWithError: . . . 8
3.4.2.2 appSettingsDidLoad: . . . 8
3.4.2.3 appSettingsDidStartDownload . . . 9
3.5 <InGameAdsDelegate> Protocol Reference . . . 9
3.5.1 Member Function Documentation . . . 9
3.5.1.1 adDidFailToGetInGameAd: . . . 9
3.5.1.2 adDidGetInGameAd: . . . 9
3.6 Spil Class Reference . . . 9
3.6.1 Detailed Description . . . 10
3.6.2 Member Function Documentation . . . 11
3.6.2.1 abtestGetTestDiff . . . 11
3.6.2.2 abtestGetTestDiffForUser: . . . 11
3.6.2.3 abtestMarkSucceedTest:withParameters: . . . 11
3.6.2.4 abtestUpdateUserInfo . . . 11
3.6.2.5 abtestUpdateUserInfoWith: . . . 11
3.6.2.6 adsCacheNextInterstitial . . . 12
3.6.2.7 adsCacheNextInterstitial: . . . 12
3.6.2.8 adsEnabled: . . . 12
3.6.2.9 adsMarkInGameAdAsShown: . . . 12
3.6.2.10 adsNextInterstitial . . . 12
3.6.2.11 adsNextInterstitial: . . . 12
3.6.2.12 adsRequestInGameAd: . . . 12
3.6.2.13 adsRequestInGameAd:atLocation: . . . 13
3.6.2.14 adsRequestInGameAdAsset: . . . 13
3.6.2.15 adsRequestInGameAdAsset:atLocation: . . . 13
3.6.2.16 adsShowMoreGames . . . 13
3.6.2.17 getABTest: . . . 14
3.6.2.18 getAds: . . . 14
3.6.2.19 getExtendedTracking: . . . 14
3.6.2.20 getSettings: . . . 14
3.6.2.21 setABTestDelegate: . . . 14
3.6.2.22 setAdsDelegate: . . . 14
3.6.2.23 setAppSettingsDelegate: . . . 15
3.6.2.24 setExtendedTrackingDelegate: . . . 15
3.6.2.25 setInGameAdsDelegate: . . . 15
3.6.2.26 sharedInstance . . . 15
3.6.2.27 spilWithAppID:token:configs: . . . 15

3.6.2.28 trackAge: . . . 16
3.6.2.29 trackEndTimedEvent: . . . 16
3.6.2.30 trackEndTimedEvent:withParams: . . . 16
3.6.2.31 trackError:message:exception: . . . 16
3.6.2.32 trackEvent: . . . 17
3.6.2.33 trackEvent:action:label:value: . . . 17
3.6.2.34 trackEvent:withParams: . . . 17
3.6.2.35 trackGender: . . . 17
3.6.2.36 trackLatitude:longitude:horizontalAccuracy:verticalAccuracy: . . . 17
3.6.2.37 trackPage: . . . 18
3.6.2.38 trackStartGestureScreen: . . . 18
3.6.2.39 trackStopGestureScreen . . . 18
3.6.2.40 trackTimedEvent: . . . 18
3.6.2.41 trackUserID: . . . 18
3.7 SpilHelpers Class Reference . . . 19
3.7.1 Detailed Description . . . 19
3.7.2 Member Function Documentation . . . 19
3.7.2.1 connectionWithURL:onSuccess:onFailure: . . . 19
3.7.2.2 getAppName . . . 19
3.7.2.3 getAppVersion . . . 19
3.7.2.4 getCountryCode . . . 20
3.7.2.5 getDeviceModel . . . 20
3.7.2.6 getDevicePlatform . . . 20
3.7.2.7 getLanguage . . . 20
3.7.2.8 getMacAddress . . . 20
3.7.2.9 getOSVersion . . . 21
3.7.2.10 getPixelRatio . . . 21
3.7.2.11 getScreenHeight . . . 21
3.7.2.12 getScreenWidth . . . 21
3.7.2.13 getUDID . . . 21
3.8 <TrackingExtendedDelegate> Protocol Reference . . . 21
3.8.1 Detailed Description . . . 22
3.8.2 Member Function Documentation . . . 22
3.8.2.1 trackExtendedDidStart . . . 22
3.8.2.2 trackExtendedDidStop . . . 22

Chapter 1

Deprecated List

Member [Spil getABTest:]

- In favor of better name conventions Method to set the ABTestDelegate and receive the proper notifications from it.

Member [Spil getAds:]

- In favor of better name conventions Method to set the AdsDelegate and receive the proper notifications from it. Also allows to receive the push notifications.

Member [Spil getExtendedTracking:]

- In favor of better name conventions Method to set up the delegate for the extended tracking events. This delegate is optional, but its usage it's encouraged since this will guarantee the calls made are actually efective and not dropped because the extended tracking wasn't started yet.

Member [Spil getSettings:]

- In favor of better name conventions Method to retrieve the AppSettings for this app. A delegate is required in order to deliver the settings downloaded from the server or loaded from the default files.

Member [SpilHelpers getMacAddress]

Gets the MAC address of this device. The address is returned as a upper case hex string without spaces or colons

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

ABTestAdapter 5
<ABTestDelegate> 5
<AdsDelegate> 6
<AppSettingsDelegate> 8
<InGameAdsDelegate> 9
Spil 9
SpilHelpers 19
<TrackingExtendedDelegate> 21

Chapter 3

Class Documentation

3.1 ABTestAdapter Class Reference

Public Member Functions

- (id) - **initWithSettings:**
- (BOOL) - **startSession**
- (void) - **endSession**
- (void) - **sendQueue:**
- (void) - **saveQueue**
- (void) - **userUpdate:**
- (void) - **event:withParams:**
- (void) - **getUserDiffs:delegate:**

The documentation for this class was generated from the following file:

- /Users/ignacio/Documents/native-unity-plugin/libs/ios-api/src/Classes/abtest/ABTestAdapter.h

3.2 <ABTestDelegate> Protocol Reference

```
#import <ABTestDelegate.h>
```

Public Member Functions

- (void) - abtestSessionDidStart
- (void) - abtestSessionDidEnd
- (void) - abtestSessionDiffReceived:

3.2.1 Detailed Description

Protocol to listen the events trigerred by the A/B Testing subsystem

3.2.2 Member Function Documentation

3.2.2.1 - (void) abtestSessionDidEnd

Method to call back after the a/b test subsystem is successfully ended.

3.2.2.2 - (void) abtestSessionDidStart

Method to call back after the a/b test subsystem is successfully started.

3.2.2.3 - (void) abtestSessionDiffReceived: (NSArray *) *diffs*

Method to call back after the a/b test subsystem receive the differences to apply over the original version. The differences come expressed as an array of objects. These objects are represented as dictionaries, where, always are defined the following keys:

- **uid**: an ID for this resource to test. A resource can contain many elements to test. Details in the next entry.
- **diff**: a dictionary with all the changes to apply to this resource. In this resource, many elements could be changed, for each element, an entry will appear in this dictionary. Each of this entry will contain a dictionary with exactly 2 keys: "new" and "old", refering to the original and value to replace with.
- **item_class**: unused for the moment.

The documentation for this protocol was generated from the following file:

- /Users/ignacio/Documents/native-unity-plugin/libs/ios-api/src/Classes/abtest/ABTestDelegate.h

3.3 <AdsDelegate> Protocol Reference

```
#import <AdsDelegate.h>
```

Public Member Functions

- (void) - adDidStart
- (void) - adDidFailToStart:
- (void) - adWillAppear
- (void) - adDidAppear
- (void) - adDidFailToAppear:
- (void) - adMoreGamesWillAppear
- (void) - adMoreGamesDidAppear
- (void) - adMoreGamesDidFailToAppear:
- (void) - adPopupDidDismiss
- (void) - adMoreGamesDidDismiss

3.3.1 Detailed Description

Protocol to handle the events triggered by the ad subsystem.

3.3.2 Member Function Documentation

3.3.2.1 - (void) adDidAppear

Method to call back after the ad is displayed. This method is only called if the ads are enabled to be displayed (enableAds:YES).

3.3.2.2 - (void) adDidFailToAppear: (NSError *) *error*

Method to call back if the ad couldn't be displayed due to any reason.

Parameters

error	The reason why the ad failed to be displayed.

3.3.2.3 - (void) adDidFailToStart: (NSError *) *error*

Method to call back after if the ad subsystem couldn't be started due to any reason.

Parameters

error	The reason why the ad subsystem failed to start.

3.3.2.4 - (void) adDidStart

Method to call back after the ad subsystem is successfully started.

3.3.2.5 - (void) adMoreGamesDidAppear

Method to call back after the more games' screen is displayed.

3.3.2.6 - (void) adMoreGamesDidDismiss

Method to call back if the more games popup showed was dismissed.

3.3.2.7 - (void) adMoreGamesDidFailToAppear: (NSError *) *error*

Method to call back if the more games' screen couldn't be displayed due to any reason.

Parameters

error	The reason why the more games' screen failed to be displayed.

3.3.2.8 - (void) adMoreGamesWillAppear

Method to call back before the next more games' screen is going to be shown.

3.3.2.9 - (void) adPopupDidDismiss

Method to call back if the ad popup showed was dismissed.

3.3.2.10 - (void) adWillAppear

Method to call back before the next ad is going to be displayed. This method is called every time the timer reach 0, regardless if the ad should be shown or not (enableAds is set to NO).

The documentation for this protocol was generated from the following file:

- /Users/ignacio/Documents/native-unity-plugin/libs/ios-api/src/Classes/ads/AdsDelegate.h

3.4 <AppSettingsDelegate> Protocol Reference

`#import <AppSettingsDelegate.h>`

Public Member Functions

- (void) - appSettingsDidLoad:
- (void) - appSettingsDidFailWithError:
- (void) - appSettingsDidStartDownload

3.4.1 Detailed Description

Protocol to handle the responses from the App Settings subsystem

3.4.2 Member Function Documentation

3.4.2.1 - (void) appSettingsDidFailWithError: (NSError *) *error* [required]

Method to call back in case the settings couldn't be loaded. Usually the reasons to call this method will be:

- if there is any parsing error in the remote settings and in the local settings.
- if there is a connection error, and the file of the defaults can be found locally.

Parameters

error	Error describing what was wrong.

3.4.2.2 - (void) appSettingsDidLoad: (NSDictionary *) *settings* [required]

Method to call back when the settings are finally loaded. This methods will receive the settings loaded in the form of a dictionary. The developers should know the structure of the dictionary since they created the default settings file.

Parameters

settings	The settings loaded. The format and the values are defined by the developer of the app.

3.4.2.3 - (void) appSettingsDidStartDownload [required]

Method to call back when the download of the settings has been started.

The documentation for this protocol was generated from the following file:

- /Users/ignacio/Documents/native-unity-plugin/libs/ios-api/src/Classes/settings/AppSettingsDelegate.h

3.5 <InGameAdsDelegate> Protocol Reference

Public Member Functions

- (void) - adDidGetInGameAd:
- (void) - adDidFailToGetInGameAd:

3.5.1 Member Function Documentation

3.5.1.1 - (void) adDidFailToGetInGameAd: (NSError *) *error*

Method to callback when an error happened trying to retrieve the ad from the server.

Parameters

error	An error code describing the cause of the error.

3.5.1.2 - (void) adDidGetInGameAd: (UIView *) *image*

Method to callback when a in game ad has been retrieved from the server. This view will handle the display event and will mark the advert as shown.

Parameters

image	A UIView that will respond to the events when it's displayed and clicked.

The documentation for this protocol was generated from the following file:

- /Users/ignacio/Documents/native-unity-plugin/libs/ios-api/src/Classes/ads/ingame/InGameAdsDelegate.h

3.6 Spil Class Reference

```
#import <Spil.h>
```

Public Member Functions

- (void) - getSettings:
- (void) - setAppSettingsDelegate:
- (void) - getAds:
- (void) - setAdsDelegate:
- (void) - adsEnabled:
- (void) - adsShowMoreGames
- (void) - adsNextInterstitial
- (void) - adsNextInterstitial:
- (void) - adsCacheNextInterstitial
- (void) - adsCacheNextInterstitial:
- (void) - setInGameAdsDelegate:
- (BOOL) - adsRequestInGameAd:
- (BOOL) - adsRequestInGameAd:atLocation:
- (void) - adsRequestInGameAdAsset:
- (void) - adsRequestInGameAdAsset:atLocation:
- (void) - adsMarkInGameAdAsShown:
- (void) - getExtendedTracking:
- (void) - setExtendedTrackingDelegate:
- (void) - trackPage:
- (void) - trackEvent:
- (void) - trackEvent:action:label:value:
- (void) - trackEvent:withParams:
- (void) - trackTimedEvent:
- (void) - trackEndTimedEvent:withParams:
- (void) - trackEndTimedEvent:
- (void) - trackError:message:exception:
- (void) - trackUserID:
- (void) - trackAge:
- (void) - trackGender:
- (void) - trackLatitude:longitude:horizontalAccuracy:verticalAccuracy:
- (void) - trackStartGestureScreen:
- (void) - trackStopGestureScreen
- (void) - getABTest:
- (void) - setABTestDelegate:
- (void) - abtestUpdateUserInfo
- (void) - abtestUpdateUserInfoWith:
- (void) - abtestGetTestDiff
- (void) - abtestGetTestDiffForUser:
- (void) - abtestMarkSucceedTest:withParameters:

Static Public Member Functions

- (Spil *) + spilWithAppID:token:configs:
- (Spil *) + sharedInstance

3.6.1 Detailed Description

Public interface to the funcionalities of the Spil iOS framework.

3.6.2 Member Function Documentation

3.6.2.1 - (void) abtestGetTestDiff

Send a request to retrieve the test differences for this user (MAC Address). The differences will be sent asynchronously to the ABTestDelegate implemented and set in the getABTest method.

3.6.2.2 - (void) abtestGetTestDiffForUser: (NSString ∗) *user*

For development purposes only. Send a request to retrieve the test differences for this user. The differences will be sent asynchronously to the ABTestDelegate implemented and set in the getABTest method.

Parameters

user	The user to force the different variants of the A/B test.

3.6.2.3 - (void) abtestMarkSucceedTest: (NSString ∗) *name* withParameters:(NSDictionary ∗) *params*

Mark a particular resource as a success with the parameters that lead to that success. This method should be called with the exact name of the resource and also it must be called with the control version to be able to compare results.

Parameters

name	The name of the resource to mark as a successful one.
params	A dictionary with extra parameters relevant for the analysis of the action called.

3.6.2.4 - (void) abtestUpdateUserInfo

Update the user basic information to create segments and improve A/B tests. This method will send:

- Country
- Language
- Device (ipad, ipod, iphone)
- OS Platform (version)

3.6.2.5 - (void) abtestUpdateUserInfoWith: (NSDictionary ∗) *extraInfo*

Update the user information to create segments and improve A/B tests. This method will send the basic information plus all the information included in the extra info

Parameters

extraInfo	A dictionary with all the extra parameter we want to submit.

See Also

- abtestUpdateUserInfo

3.6.2.6 - (void) adsCacheNextInterstitial

Cache the next interstitial image to speed up the load time. Uses the default location.

3.6.2.7 - (void) adsCacheNextInterstitial: (NSString ∗) *location*

Cache the next interstitial image to speed up the load time. Uses the specified location.

Parameters

location	Location to be used for this interstitial

3.6.2.8 - (void) adsEnabled: (BOOL) *state*

Turn on/off if the ads should be displayed. The ads are displayed by default. For gameplay screens should be turned off. After return to the menus should be turned on again.

Parameters

state	Indicates if the ads should be displayed or not.

3.6.2.9 - (void) adsMarkInGameAdAsShown: (NSString ∗) *adId*

Mark the ad as shown, this will be use only for the unity plugin.

Parameters

adId	The advert id returned by chartboost assets lib.

3.6.2.10 - (void) adsNextInterstitial

Shows an ad right away, using the default location.

3.6.2.11 - (void) adsNextInterstitial: (NSString ∗) *location*

Shows an ad right away, using the specified location.

Parameters

location	Location to be used for this interstitial

3.6.2.12 - (BOOL) adsRequestInGameAd: (CGSize) *size*

Makes a request to get an advert and return it to the invoker when it's done through the AdsDelegate implementation set up prior the call to this method. This methods returns right away and gives NO if there is not a valid chartboost instance, YES otherwise. Although this method returns YES, it doesn't mean the ad will be in fact retrieved and returned.

Parameters

size	The width and height desired for this ad.

Returns

NO if there chartboost provider is not valid. YES otherwise.

3.6.2.13 - (BOOL) adsRequestInGameAd: (CGSize) *size* atLocation:(NSString ∗) *location*

Makes a request to get an advert and return it to the invoker when it's done through the AdsDelegate implementation set up prior the call to this method. This methods returns right away and gives NO if there is not a valid chartboost instance, YES otherwise. Although this method returns YES, it doesn't mean the ad will be in fact retrieved and returned.

Parameters

size	The width and height desired for this ad.
location	Location to be used for this interstitial

Returns

NO if there chartboost provider is not valid. YES otherwise.

3.6.2.14 - (void) adsRequestInGameAdAsset: (NSDictionary ∗) (NSError ∗) *callback*

Retrieves the JSON description of the assets, and pass it back to the invoker block a decoded JSON format. The invoker is the responsable for scale and download the asset if it's present

Parameters

callback	A callback that will receive the information as a dictionary that can be serialized.

3.6.2.15 - (void) adsRequestInGameAdAsset: (NSDictionary ∗) (NSError ∗) *callback* atLocation:(NSString ∗) *location*

Retrieves the JSON description of the assets, and pass it back to the invoker block a decoded JSON format. The invoker is the responsable for scale and download the asset if it's present

Parameters

callback	A callback that will receive the information as a dictionary that can be serialized.
location	Location to be used for this interstitial

3.6.2.16 - (void) adsShowMoreGames

Force to show the More Games screen.

3.6.2.17 - (void) getABTest: (id< ABTestDelegate >) *delegate*

Deprecated • In favor of better name conventions Method to set the ABTestDelegate and receive the proper notifications from it.

Parameters

delegate	The delegate to handle the events generated by the A/B test subsystem.

3.6.2.18 - (void) getAds: (id< AdsDelegate >) *delegate*

Deprecated • In favor of better name conventions Method to set the AdsDelegate and receive the proper notifications from it. Also allows to receive the push notifications.

Parameters

delegate	The delegate to handle the events generated by the Ads subsystem.

3.6.2.19 - (void) getExtendedTracking: (id< TrackingExtendedDelegate >) *delegate*

Deprecated • In favor of better name conventions Method to set up the delegate for the extended tracking events. This delegate is optional, but its usage it's encouraged since this will guarantee the calls made are actually efective and not dropped because the extended tracking wasn't started yet.

3.6.2.20 - (void) getSettings: (id< AppSettingsDelegate >) *delegate*

Deprecated • In favor of better name conventions Method to retrieve the AppSettings for this app. A delegate is required in order to deliver the settings downloaded from the server or loaded from the default files.

Parameters

delegate	The delegate to handle the response of the AppSettings subsystem.

3.6.2.21 - (void) setABTestDelegate: (id< ABTestDelegate >) *delegate*

Method to set the ABTestDelegate and receive the proper notifications from it.

Parameters

delegate	The delegate to handle the events generated by the A/B test subsystem.

3.6.2.22 - (void) setAdsDelegate: (id< AdsDelegate >) *delegate*

Method to set the AdsDelegate and receive the proper notifications from it. Also allows to receive the push notifications.

Parameters

delegate	The delegate to handle the events generated by the Ads subsystem.

3.6.2.23 - (void) setAppSettingsDelegate: (id< AppSettingsDelegate >) *delegate*

Method to retrieve the AppSettings for this app. A delegate is required in order to deliver the settings downloaded from the server or loaded from the default files.

Parameters

delegate	The delegate to handle the response of the AppSettings subsystem.

3.6.2.24 - (void) setExtendedTrackingDelegate: (id< TrackingExtendedDelegate >) *delegate*

Method to set up the delegate for the extended tracking events. This delegate is optional, but its usage it's encouraged since this will guarantee the calls made are actually efective and not dropped because the extended tracking wasn't started yet.

3.6.2.25 - (void) setInGameAdsDelegate: (id< InGameAdsDelegate >) *delegate*

Set the delegate to handle the events received by the in game ads system.

Parameters

delegate	The ads delegate who is going to handle the events.

3.6.2.26 + (Spil*) sharedInstance

Method to retrieve the sharedInstance, since this class is a singleton. The instance returned could be nil if the constructor above haven't been called or if was error occured.

Returns

The shared instance of this Spil object.

3.6.2.27 + (Spil*) spilWithAppID: (NSString *) *applicationID* token:(NSString *) *authenticationToken* configs:(NSDictionary *) *configurations*

Create a Spil object singleton with an application ID and authentication token that will be used along the framework for multiple services. Also you must specify some configurations to control the behaivor of the framework. Most important, if the framework should act like a development environment or a production environment. Some validations about the configurations are made, if one fails, a nil object is returned, and the error is written in the console log.

Parameters

applicationID	The application ID provided by Spil Games, it can't be nil.
authentication-Token	The authentication token provided by Spil Games, it can't be nil.
configurations	A dictionary with the posible settings to be used by spil framework.

Returns

The Spil object that will be use for further calls.

3.6.2.28 - (void) trackAge: (int) *age*

Track the age of the user is logged in the application. Useful for demographic information.

Parameters

age	The age to track.

3.6.2.29 - (void) trackEndTimedEvent: (NSString ∗) *event*

Track requesto to register the end of an event that was started. If the tracking system support it natively the equivalent method will be used. Otherwise, an event with the end timestamp(epoch) is issued.

Parameters

event	The event to track. Should match with the starting event.

See Also

- trackTimedEvent:

3.6.2.30 - (void) trackEndTimedEvent: (NSString ∗) *event* withParams:(NSDictionary ∗) *params*

Track requesto to register the end of an event that was started. If the tracking system support it natively the equivalent method will be used. Otherwise, an event with the end timestamp(epoch) is issued.

Parameters

event	The event to track. Should match with the starting event.
params	The parameters when the event was finished.

See Also

- trackTimedEvent:

3.6.2.31 - (void) trackError: (NSString ∗) *event* message:(NSString ∗) *msg* exception:(NSException ∗) *exception*

Track request to register an error that has occured in the application.

Parameters

event	The event to track.
msg	The message with the detail of the error.
exception	The exception that causes the error.

3.6.2.32 - (void) trackEvent: (NSString ∗) *event*

Track request to register an event. The event could be actions taken on some object like unlocking an achievement, or a getting a hi score. In google analytics jargot they will be register as a "category"="event" and the event passed as parameter as the action taken. In flurry jargot, an event with parameters is created and the parameters match with google analytics (category,action,label,value)

Parameters

event	The event to track.

3.6.2.33 - (void) trackEvent: (NSString ∗) *category* action:(NSString ∗) *action* label:(NSString ∗) *label* value:(int) *value*

Track request to register an event under a particular category. The parameters match with the google analytics' ones. For flurry, a event with parameters is issued.

Parameters

category	The category for this event.
action	The action took on that category.
label	Optional. A string label to especify something about the action.
value	Optional. A integer value. useful to especify error codes.

3.6.2.34 - (void) trackEvent: (NSString ∗) *event* withParams:(NSDictionary ∗) *params*

Track request to register an event with particular parameters. This match the Flurry's logEvent:withParameters:. It's not supported for GAN.

Parameters

event	The event to track
params	Additional parameters to attach to the event.

3.6.2.35 - (void) trackGender: (BOOL) *male*

Track the gender of the user is logged in the application. Useful for demographic information.

Parameters

male	YES|TRUE if the player is male, NO|FALSE if the player is female.

3.6.2.36 - (void) trackLatitude: (CLLocationDegrees) *latitude* longitude:(CLLocationDegrees) *longitude* horizontalAccuracy:(CLLocationAccuracy) *hAccuracy* verticalAccuracy:(CLLocationAccuracy) *vAccuracy*

Track the location information of the player if it's available. If the tracking system support it natively the equivalent method will be used. Otherwise, an event with the information of the location (latitude, longitude and accuracy) is issued.

Parameters

latitude	The latitude where the device is. (it's a double value)
longitude	The longitude where the device is. (it's a double value).
hAccuracy	The horizontal accuracy of the measurement. (it's a double value).
vAccuracy	The vertical accuracy of the measurement. (it's a double value).

3.6.2.37 - (void) trackPage: (NSString ∗) *page*

Track request to register a particular page. It can be used to keep track of the current screen separetely from the events. If the session is not started yet, this request is ignored.

Parameters

page	The page name/url to track.

3.6.2.38 - (void) trackStartGestureScreen: (NSString ∗) *screenName*

Start the recording of the gestures for a new screen, the gestures for this screen will be stored together regarding how many times this screen has been started.

Parameters

screenName	The name of the screen to record.

3.6.2.39 - (void) trackStopGestureScreen

Stop the recording gestures for this screen, the gestures are drop until you start a new screen.

3.6.2.40 - (void) trackTimedEvent: (NSString ∗) *event*

Track requesto to register the start of an event that should end in a finite time. If the tracking system support it natively the equivalent method will be used. Otherwise, an event with the start timestamp(epoch) is issued.

Parameters

event	The event to track.

3.6.2.41 - (void) trackUserID: (NSString ∗) *userID*

Track the user ID that logged in the application. Useful to keep track of how many users return to the application.

Parameters

userID	The user ID to track.

The documentation for this class was generated from the following file:

- /Users/ignacio/Documents/native-unity-plugin/libs/ios-api/src/Classes/Spil.h

3.7 SpilHelpers Class Reference

`#import <SpilHelpers.h>`

Static Public Member Functions

- (NSString ∗) + getMacAddress
- (NSString ∗) + getUDID
- (NSString ∗) + getAppVersion
- (NSString ∗) + getAppName
- (NSString ∗) + getLanguage
- (NSString ∗) + getCountryCode
- (NSString ∗) + getDeviceModel
- (NSString ∗) + getDevicePlatform
- (NSString ∗) + getOSVersion
- (int) + getScreenWidth
- (int) + getScreenHeight
- (float) + getPixelRatio
- (void) + connectionWithURL:onSuccess:onFailure:

3.7.1 Detailed Description

Helper class to retrieve information about information relative to the device.

3.7.2 Member Function Documentation

3.7.2.1 + (void) connectionWithURL: (NSURL ∗) *url* onSuccess:(NSData ∗) *successCallback* onFailure:(NSError ∗) *failureCallback*

Simplest method to start a GET connection to the URL provided as parameter. The responses, successful or not, are notified to the respective blocks (if any).

Parameters

url	The URL to send the request. This URL can contain extra parameters in the correct query string form.
onSuccess	Block to notify the data retrieved if everything was ok.
onFailure	Block to notify the error.

3.7.2.2 + (NSString∗) getAppName

Gets the current name of the app from the Info.plist

Returns

The CFBundleName entry in the Info.plist

3.7.2.3 + (NSString∗) getAppVersion

Gets the current version of the app from the Info.plist

Returns

The CFBundleVersion entry in the Info.plist

3.7.2.4 + (NSString∗) getCountryCode

Gets the current country configured in the device.

Returns

The short country code configured in the device.

3.7.2.5 + (NSString∗) getDeviceModel

Gets the device model (ipod, iphone, ipad, simulator, etc)

Returns

The device model.

3.7.2.6 + (NSString∗) getDevicePlatform

Gets the device platform family (iphone, ipad)

Returns

The device family.

3.7.2.7 + (NSString∗) getLanguage

Gets the current preferred language for the OS.

Returns

The short version of the preferred language.

3.7.2.8 + (NSString∗) getMacAddress

Deprecated Gets the MAC address of this device. The address is returned as a upper case hex string without spaces or colons

Returns

The MAC address of this device.

3.7.2.9 + (NSString*) getOSVersion

Gets the OS version installed in the device.

Returns

The OS version running in the device.

3.7.2.10 + (float) getPixelRatio

Gets the screen pixel ratio. A ratio 1-to-1 means a traditional screen. A ration 2-to-1 means a retina display.

Returns

The screen ratio.

3.7.2.11 + (int) getScreenHeight

Gets the screen height on points.

Returns

The screen height on points.

3.7.2.12 + (int) getScreenWidth

Gets the screen width on points.

Returns

The screen width on points.

3.7.2.13 + (NSString*) getUDID

Gets a new UDID based on OpenUDID implementation

Returns

A UDID generated by OpenUDID which will be shared across all apps which implements OpenUDID.

The documentation for this class was generated from the following file:

- /Users/ignacio/Documents/native-unity-plugin/libs/ios-api/src/Classes/SpilHelpers.h

3.8 <TrackingExtendedDelegate> Protocol Reference

```
#import <TrackingExtendedDelegate.h>
```

Public Member Functions

- (void) - trackExtendedDidStart
- (void) - trackExtendedDidStop

3.8.1 Detailed Description

Protocol to handle the responses from the Extended Tracking subsystem

3.8.2 Member Function Documentation

3.8.2.1 - (void) trackExtendedDidStart

Method to call back when the any of the extended trackers are started. If the camera tracker is set up this method is called after the confirmation pop up is done, and if there is at least one tracker active. When the camera tracker is not set up, this method is called when any of the other are activated.

3.8.2.2 - (void) trackExtendedDidStop

Method to call back when ALL the extended trackers are stopped. This is an informative call.

The documentation for this protocol was generated from the following file:

- /Users/ignacio/Documents/native-unity-plugin/libs/ios-api/src/Classes/tracking/TrackingExtendedDelegate.h

Index

<ABTestDelegate>, 5
<AdsDelegate>, 6
<AppSettingsDelegate>, 8
<InGameAdsDelegate>, 9
<TrackingExtendedDelegate>, 21

ABTestAdapter, 5
ABTestDelegate-p
    abtestSessionDidEnd, 6
    abtestSessionDidStart, 6
    abtestSessionDiffReceived:, 6
abtestGetTestDiff
    Spil, 11
abtestGetTestDiffForUser:
    Spil, 11
abtestMarkSucceedTest:withParameters:
    Spil, 11
abtestSessionDidEnd
    ABTestDelegate-p, 6
abtestSessionDidStart
    ABTestDelegate-p, 6
abtestSessionDiffReceived:
    ABTestDelegate-p, 6
abtestUpdateUserInfo
    Spil, 11
abtestUpdateUserInfoWith:
    Spil, 11
adDidAppear
    AdsDelegate-p, 7
adDidFailToAppear:
    AdsDelegate-p, 7
adDidFailToGetInGameAd:
    InGameAdsDelegate-p, 9
adDidFailToStart:
    AdsDelegate-p, 7
adDidGetInGameAd:
    InGameAdsDelegate-p, 9
adDidStart
    AdsDelegate-p, 7
adMoreGamesDidAppear
    AdsDelegate-p, 7
adMoreGamesDidDismiss
    AdsDelegate-p, 7
adMoreGamesDidFailToAppear:
    AdsDelegate-p, 7
adMoreGamesWillAppear
    AdsDelegate-p, 7
adPopupDidDismiss
    AdsDelegate-p, 7
adWillAppear
    AdsDelegate-p, 8
adsCacheNextInterstitial
    Spil, 11
adsCacheNextInterstitial:
    Spil, 12
AdsDelegate-p
    adDidAppear, 7
    adDidFailToAppear:, 7
    adDidFailToStart:, 7
    adDidStart, 7
    adMoreGamesDidAppear, 7
    adMoreGamesDidDismiss, 7
    adMoreGamesDidFailToAppear:, 7
    adMoreGamesWillAppear, 7
    adPopupDidDismiss, 7
    adWillAppear, 8
adsEnabled:
    Spil, 12
adsMarkInGameAdAsShown:
    Spil, 12
adsNextInterstitial
    Spil, 12
adsNextInterstitial:
    Spil, 12
adsRequestInGameAd:
    Spil, 12
adsRequestInGameAd:atLocation:
    Spil, 13
adsRequestInGameAdAsset:
    Spil, 13
adsRequestInGameAdAsset:atLocation:
    Spil, 13
adsShowMoreGames
    Spil, 13
AppSettingsDelegate-p
    appSettingsDidFailWithError:, 8
    appSettingsDidLoad:, 8
    appSettingsDidStartDownload, 9
appSettingsDidFailWithError:
    AppSettingsDelegate-p, 8
appSettingsDidLoad:

AppSettingsDelegate-p, 8
appSettingsDidStartDownload
AppSettingsDelegate-p, 9

connectionWithURL:onSuccess:onFailure:
SpilHelpers, 19

getABTest:
Spil, 13
getAds:
Spil, 14
getAppName
SpilHelpers, 19
getAppVersion
SpilHelpers, 19
getCountryCode
SpilHelpers, 20
getDeviceModel
SpilHelpers, 20
getDevicePlatform
SpilHelpers, 20
getExtendedTracking:
Spil, 14
getLanguage
SpilHelpers, 20
getMacAddress
SpilHelpers, 20
getOSVersion
SpilHelpers, 20
getPixelRatio
SpilHelpers, 21
getScreenHeight
SpilHelpers, 21
getScreenWidth
SpilHelpers, 21
getSettings:
Spil, 14
getUDID
SpilHelpers, 21

InGameAdsDelegate-p
adDidFailToGetInGameAd:, 9
adDidGetInGameAd:, 9

setABTestDelegate:
Spil, 14
setAdsDelegate:
Spil, 14
setAppSettingsDelegate:
Spil, 15
setExtendedTrackingDelegate:
Spil, 15
setInGameAdsDelegate:
Spil, 15
sharedInstance
Spil, 15
Spil, 9
abtestGetTestDiff, 11
abtestGetTestDiffForUser:, 11
abtestMarkSucceedTest:withParameters:, 11
abtestUpdateUserInfo, 11
abtestUpdateUserInfoWith:, 11
adsCacheNextInterstitial, 11
adsCacheNextInterstitial:, 12
adsEnabled:, 12
adsMarkInGameAdAsShown:, 12
adsNextInterstitial, 12
adsNextInterstitial:, 12
adsRequestInGameAd:, 12
adsRequestInGameAd:atLocation:, 13
adsRequestInGameAdAsset:, 13
adsRequestInGameAdAsset:atLocation:, 13
adsShowMoreGames, 13
getABTest:, 13
getAds:, 14
getExtendedTracking:, 14
getSettings:, 14
setABTestDelegate:, 14
setAdsDelegate:, 14
setAppSettingsDelegate:, 15
setExtendedTrackingDelegate:, 15
setInGameAdsDelegate:, 15
sharedInstance, 15
spilWithAppID:token:configs:, 15
trackAge:, 16
trackEndTimedEvent:, 16
trackEndTimedEvent:withParams:, 16
trackError:message:exception:, 16
trackEvent:, 16
trackEvent:action:label:value:, 17
trackEvent:withParams:, 17
trackGender:, 17
trackLatitude:longitude:horizontalAccuracy:verticalAccuracy:, 17
trackPage:, 18
trackStartGestureScreen:, 18
trackStopGestureScreen, 18
trackTimedEvent:, 18
trackUserID:, 18
SpilHelpers, 19
connectionWithURL:onSuccess:onFailure:, 19
getAppName, 19
getAppVersion, 19
getCountryCode, 20
getDeviceModel, 20
getDevicePlatform, 20
getLanguage, 20
getMacAddress, 20
getOSVersion, 20

getPixelRatio, 21
getScreenHeight, 21
getScreenWidth, 21
getUDID, 21
spilWithAppID:token:configs:
Spil, 15

trackAge:
Spil, 16
trackEndTimedEvent:
Spil, 16
trackEndTimedEvent:withParams:
Spil, 16
trackError:message:exception:
Spil, 16
trackEvent:
Spil, 16
trackEvent:action:label:value:
Spil, 17
trackEvent:withParams:
Spil, 17
trackExtendedDidStart
TrackingExtendedDelegate-p, 22
trackExtendedDidStop
TrackingExtendedDelegate-p, 22
trackGender:
Spil, 17
trackLatitude:longitude:horizontalAccuracy:vertical-
Accuracy:
Spil, 17
trackPage:
Spil, 18
trackStartGestureScreen:
Spil, 18
trackStopGestureScreen
Spil, 18
trackTimedEvent:
Spil, 18
trackUserID:
Spil, 18
TrackingExtendedDelegate-p
trackExtendedDidStart, 22
trackExtendedDidStop, 22