Saturday, 2 April 2016

POKKT SDK v4.1 Integration Guide for Xamarin (iOS)

Saturday, 2 April 2016

Contents:

1. Overview
2. Configuration steps
3. Implementation steps
4. Debugging and Logging
5. Important Points

1. Overview

Thank you for choosing Pokkt SDK Plugin v4.1 for Xamarin. Pokkt SDK supports Rewarded Ad/Non-Rewarded Ad campaigns feature. This document contains all the information that is needed by you to setup the SDK with your project. The current plugin supports mediation for various third party ad-networks. These are:

- AdColony
- AppLovon
- Chartboost
- Fyber
- SuperSonic
- UnityAds
- Vungle
- Tapjoy
- Facebook
- AdMob

A separate set of documents is provided for each of these, explaining the implementation process.

There is a sample app provided with the SDK. We will be referencing this app during the course of explanation in this document. It is suggested that you should check that app to understand the following process in detail.

2. Configuration Steps

All we need is the file provided: *PluginExtension.zip.* This zip file contains three files one is *PokktExtension.dll, PokktExtension.iOS.dll* which you need to add it in project reference and 3rd is *PokktSDKResource.bundle*.

Steps need to follow:

Step 1:

1. Add the *PokktExtension.dll and PokktExtension.iOS.dll* to your project's Reference directory.

2. Extract the contents of *PokktSDKResource.bundle* folder and copy them in resource folders in your project.

Step 2:

In order to use PokktSDK's background fetch functionality, enable "*Info.plist -> Background Modes ->Background Fetch*". Then write the following code-snippet inside "*FinishedLaunching*" method of *AppDelegate.cs* class.

```
application.SetMinimumBackgroundFetchInterval(UIApplication.BackgroundFetchIntervalMinimum);
```

Step 3:

In order to enable local notifications for InApp Notifications, mention the following inside "didFinishLaunchingWithOptions" method of the app-delegate class:

```
UIUserNotificationSettings settings = UIUserNotificationSettings.GetSettingsForTypes((UIUserNotificationType.Badge  | UIUserNotificationType.Alert | UIUserNotificationType.Sound), null);

application.RegisterUserNotificationSettings(settings);
```

Step 4.

Further, implement/update the background-fetch delegate methods in *AppDelegate.cs* class. Invoke "*CallBackgroundTaskCompletionHandler*" method from "*PerformFetch*". Observe the following code-snippet for reference:

```
public override void PerformFetch(UIApplication application, Action<UIBackgroundFetchResult> completionHandler)  {
PokktXamarinExtension.PokktManager.CallBackgroundTaskCompletionHandler(completionHandler);
}
```

Step 5:

Invoke "*InAppNotificationEvent*" if the user taps on local notification, do this in the "*ReceivedLocalNotification*" inside *AppDelegate.cs* class. Check the following reference:

```
public override void ReceivedLocalNotification(UIApplication application, UILocalNotification notification)    {
        PokktXamarinExtension.PokktManager.InAppNotificationEvent(notification);
    }
```

3. Implementation Steps

Common

1. For all invocation of Pokkt SDK developer will make use of methods available in *PokktManager* class. This class only have static methods.

2. You need to set extension before calling any method like below. This is mandatory to do it.
   PokktExtension.PokktManager.SetNativeExtentions(new IosExtension(this));

3. In *PokktConfig* you can set *ApplicationId*, *SecurityKey, IntegrationType*. which are must for all type of integrations. Please check the sample app.

4. Before calling any other methods from the *PokktManager,* please make sure that you have called the *InitPokkt with passing PokktConfig object*.

5. If you are doing server to server integration with pokkt you can also set *ThirdPartyUserId* in *PokktConfig*.

9. To use google analytics, please set AnalyticsType and Analytics ID in *PokktConfig*.

   SelectedAnalyticsType = AnalyticsType.GOOGLE_ANALYTICS
   GoogleAnalyticsId = "Id".

10. To use flurry analytics please set AnalyticsType and Flurry Application Key in *PokktConfig*.

    SelectedAnalyticsType = AnalyticsType.FLURRY
    FlurryApplicationKey = "key".

11. To use mix panel analytics please set AnalyticsType and Mix PanelProject Token in *PokktConfig*

    SelectedAnalyticsType = AnalyticsType.MIXPANEL
    MixPanelProjectToken = "token".

12. To use mix panel analytics please set AnalyticsType and Fabric Token in *PokktConfig*

SelectedAnalyticsType = AnalyticsType.FABRIC

12. Please call *PokktManager.NotifyAppInstall()* to send your application installation information to Pokkt.

13. Please call *PokktManager.TrackIAP(InAppPurchaseDetail)* to send any in-app purchase information to Pokkt.

```
InAppPurchaseDetails purchaseDetails = new InAppPurchaseDetails ();
purchaseDetails.CurrencyCode = "IN";
purchaseDetails.Description = "description";
purchaseDetails.Price = 10;
purchaseDetails.ProductId = “product id";
purchaseDetails.PurchaseData = "date";
purchaseDetails.PurchaseSignature = "signature";
purchaseDetails.PurchaseStore = IAPStoreType.AMAZON;
PokktManager.TrackIAP (purchaseDetails);
```

Session

1. We have option to start session for tracking: *StartSession* and *EndSession* methods in *PokktManager*.
2. You should call *StartSession* at the start of his application if you want to use this but this is the optional and call it after setting application id and security key.
3. You should call *EndSession* at the end of his application.

AdConfig

1. In *AdConfig* you should set *ScreenName* and *IsRewarded.* This screen name will be created on pokkt dashboard.
2. In *AdConfig* , developer can also set *ShouldAllowSkip, DefaultSkipTime, SkipConfirmMessage, ShouldAllowMute, ShouldSkipConfirm, SkipConfirmYesLabel, SkipConfirmNoLabel, SkipTimerMessage* and *IncentiveMessage* . These values can be used to configure the behaviour of ad.
3. If you want to enable/disable the skip button on video screen please set *ShouldAllowSkip* as true/false.The default value for S*houldAllowSkip* is true.
4. If you have enabled skipped button by setting S*houldAllowSkip* as true then you can control after how many seconds the skip button will be visible in video by setting D*efaultSkipTime* to appropriate value.Since most videos will be 30 sec or less please set *DefaultSkipTime* as 10 or less.You can also give your own skip message by setting *SkipConfirmMessage* on *AdConfig*

5. The *screenName* has default value as *default* and can be used by you to give different screen name for different places in your app where you are showing ads. You will control ad targeting based on these screen names which should match exactly with screen names defined in dashboard. ScreenName can not contain white spaces and only special characters allowed are hyphen and underscore.
6. You can choose to show ad with or without incentive to user by setting *IsRewarded* as true or false. Ad gratification will only happen for incentivised playback.
7. You can configure the ad skip dialog yes/no labels by setting *SkipConfirmYesLabel* and *SkipConfirmNoLabel*.
8. You can configure the ad incentive message by setting *IncentiveMessage*.
9. You can configure the ad skip timer message by setting *SkipTimerMessage*. The message must contain a ## placeholder to show skip time value, which will keep changing as per the time.

Rewarded Ad/Non-Rewarded Ad

1. You need to set true/false for rewarded or non-rewarded ad like below: adConfig.IsRewarded = true/false;
2. You will have to call *PokktManager.CacheAd(adConfig);* to start caching ads on device.
3. You will need to register event for getting callback for Ad related callback like below and also please check given sample app VideoActivity.cs class.

```
PokktManager.Dispatcher.PokktInitialisedEvent += PokktInitialised;
PokktManager.Dispatcher.AdCachingCompletedEvent += AdCachingCompleted;
PokktManager.Dispatcher.AdCachingFailedEvent += AdCachingFailed;
PokktManager.Dispatcher.AdAvailabilityEvent += AdAvailability;
PokktManager.Dispatcher.AdDisplayedEvent += AdDisplayed;
PokktManager.Dispatcher.AdCompletedEvent += AdCompleted;
PokktManager.Dispatcher.AdClosedEvent += AdClosed;
PokktManager.Dispatcher.AdSkippedEvent += AdSkipped;
PokktManager.Dispatcher.AdGratifiedEvent += AdGratified;
```

4. You can call *PokktManager.CheckAdAvailability(adConfig)* to check if the campaign are available for a particular adConfig before you try to show ad.
5. You can call *PokktManager.ShowAd(adConfig);* to show ad.

6. You will get different callbacks as given in *AdCampaignDelegate* implementation class for ad display.
7. Please reward user only from the *AdGratified* method.

Mediation Info

2. Pokkt SDK now supports 10 ad networks which you can integrate in your application for better monetisation.
3. To integrate these networks through Pokkt, please visit the mediation menu on downloads page and download *xamarin mediation zip* and *documentation zip* files.
4. Please follow the mediation integration documents shipped for each network.
5. You will need to create account on these networks and add the network details in your Pokkt dashboard after login into your account on pokkt website.
6. You will also need to do the mapping of Pokkt screens with the corresponding ad networks' placement id/zone id/ad unit etc in the dashboard.

Export Logs

1. Developer should call *PokktManager.ExportLog()* to export the Pokkt SDK logs to folder of your choice.
2. This API shows a folder chooser dialog where user can choose a particular folder.
3. User can also create a new folder where user wants to export the logs

Optional Parameters

PokktConfig also has provision for developers to provide extra user data available with them to pokkt. We currently support following data points: SetName, SetAge, SetSex, SetMobileNo, SetEmailAddress, SetLocation, SetBirthday, SetMaritalStatus, SetFacebookId, SetTwitterHandle, SetEducation, SetNationality, SetEmployment and SetMaturityRating.

4. Debugging and Logging

You can enable the SDK logs by setting the debugging option to true anytime.

PokktManager.SetDebug(<true/false>);

5. Important Points

- Please do not copy point from this pdf as it may introduce unwanted character and space in your code. Instead please refer to sample app source.
- Please also refer to sample app source code for better understanding of implementation.