



POKKT SDK v5.0.0 Integration Guide for AIR (iOS)

Contents:

1. Overview
2. Configuration steps
3. Implementation steps
4. Important Points

1. Overview

Thank you for choosing **Pokkt SDK Plugin v5.0.0** for **AIR**. Pokkt SDK supports Reward-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: These are:

1. AdColony
2. AppLovin
3. Chartboost
4. Fyber
5. Supersonic
6. UnityAds
7. Tapjoy
8. Vungle

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 file which contains two files PokktExtension.ane and PokktMediation.ane.

Add .ane file in your project properties and also please read [here](#) to understand how AIR ane works.

////////////////////////////////////

Note: Please do not copy the code points from this PDF file as it may introduce unwanted characters and space in your code. Instead please refer to sample app source code provided with the sample app.

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. In PokktConfig you can set ApplicationId, SecurityKey, IntegrationType. which are must for all type of integrations. Please check the sample app.
3. Before calling any other methods from the PokktManager, please make sure that you have called the InitPokkt with passing PokktConfig object and this will call only once.
4. If you are doing server to server integration with POKKT you can also set ThirdPartyUserId in PokktConfig.
5. Apart from above mentioned parameters you can assign additional ones based on your integration type (please refer to OfferWall and Reward sections below).
6. While in development, please call PokktManager.setDebug(true); to see Pokkt debug logs and toast messages. Please make sure to change this to PokktManager.setDebug(false); for production build.
7. To use google analytics, please set AnalyticsType and Analytics ID in

```
PokktConfig.pokktConfig.selectedAnalyticsType = AnalyticsType.GOOGLE_ANALYTICS;  
pokktConfig.googleAnalyticsID = "Id";
```

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

```
pokktConfig.selectedAnalyticsType = AnalyticsType.FLURRY;  
pokktConfig.flurryApplicationKey = "key";
```

9. To use mix panel analytics please set AnalyticsType and Mix PanelProject Token in PokktConfig:

```
pokktConfig.selectedAnalyticsType = AnalyticsType.FLURRY;  
pokktConfig.mixPanelProjectToken = "token";
```

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

```
InAppPurchaseDetails purchaseDetails = new InAppPurchaseDetails ();
pokktConfig.selectedAnalyticsType = AnalyticsType.FABRIC
```

11. Please call `trackIAP(InAppPurchaseDetail)` to send any in-app purchase information to Pokkt. Like below you can do this.

```
// create object of InAppPurchaseDetails and set properties for purchaseDetails
// and pass object
PokktManager.trackIAP (InAppPurchaseDetails);
```

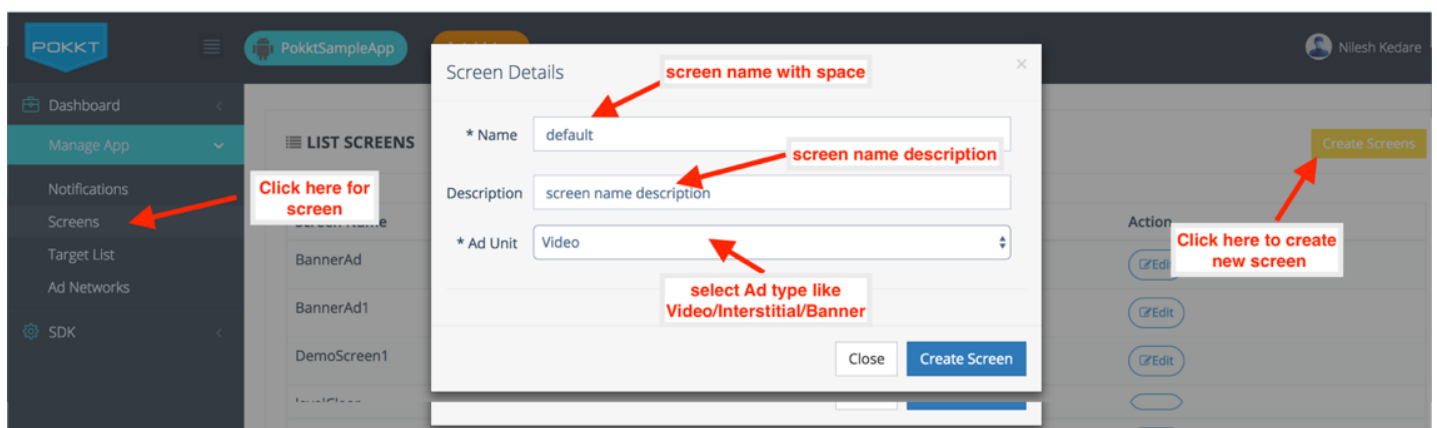
Session

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 need to provide **screenName**, **isRewarded** and **adFormat**. This screen name will be created on Pokkt dashboard.

Screen Name: For screen name, you will have to create in Pokkt dashboard. Please check below screen. This will help you to understand how to create new screen name in Pokkt dashboard.



1. In `AdConfig`, developer can also set `shouldAllowSkip`, `defaultSkipTime`, `skipConfirmMessage`, `backButtonDisabled`, `shouldAllowMute`, `shouldSkipConfirm`, `skipConfirmYesLabel`, `skipConfirmNoLabel`, `skipTimerMessage` and `incentiveMessage`. These values can be used to configure the behaviour of ad.
2. If you want to enable/disable the skip button on video screen please set `shouldAllowSkip` as `true/false`. The default value for `shouldAllowSkip` is `true`.

3. If you have enabled skipped button by setting `shouldAllowSkip` as `true` then you can control after how many seconds the skip button will be visible in video by setting `defaultSkipTime` 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`
4. 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.
5. 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.
6. You can configure the ad skip dialog yes/no labels by setting `skipConfirmYesLabel` and `skipConfirmNoLabel`.
7. You can configure the ad incentive message by setting `incentiveMessage`.
8. 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 in `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 `PlayVideoScreen` class.

```
PokktManager.dispatcher.addListener(PokktEvent.POKKT_INITIALISED_EVENT, onPokktInitialised);
PokktManager.dispatcher.addListener(PokktEvent.AD_AVAILABILITY_STATUS_EVENT, onAdAvailability);
PokktManager.dispatcher.addListener(PokktEvent.AD_CACHING_COMPLETED_EVENTS, onAdCachingCompleted);
PokktManager.dispatcher.addListener(PokktEvent.AD_CACHING_FAILED_EVENTS, onAdCachingFailed);
PokktManager.dispatcher.addListener(PokktEvent.AD_CLOSED_EVENTS, onAdClosed);
PokktManager.dispatcher.addListener(PokktEvent.AD_COMPLETED_EVENTS, onAdCompleted);
PokktManager.dispatcher.addListener(PokktEvent.AD_SKIPPED_EVENTS, onAdSkipped);
PokktManager.dispatcher.addListener(PokktEvent.AD_DISPLAYED_EVENTS, onAdDisplayed);
PokktManager.dispatcher.addListener(PokktEvent.AD_GRATIFIED_EVENTS, onAdGratified);
```

4. You can call `checkAdAvailability(adConfig)` to check if the campaign are available for a particular `adConfig` before you try to show ad.

```
PokktManager.checkAdAvailability(adConfig)
```

5. You can call `showAd` to show reward / non-reward ad.

```
PokktManager.showAd(adConfig);
```

6. Please reward user only from the `AdGratified` method.

Banner Ad

Pokkt SDK allows to show banner ad on your screen. You can set any custom size or any position for banner. There are few fixed position already given in `BannerPosition` class. But you can customise that also.

1. **Load Banner:** Use `loadBanner` to show banner ad like below:

```
PokktManager.loadBanner(ScreenName, BannerPosition.TOP_CENTER);
```

2. **Remove Banner:** Use `removeBanner` to remove banner ad from screen like below:

```
PokktManager.removeBanner(ScreenName);
```

3. **Auto Refresh Banner:** Use `setBannerAutoRefresh` method to disable or enable auto refresh. Default

it is true and it will refresh automatically based on given time on Pokkt dashboard for particular screen name.

```
PokktManager.setBannerAutoRefresh(false/true);
```

4. **Banner Position:** Use BannerPosition class properties for banner position.

5. **Custom banner:** There are also option is given to custom banner size and set custom position by using below method:

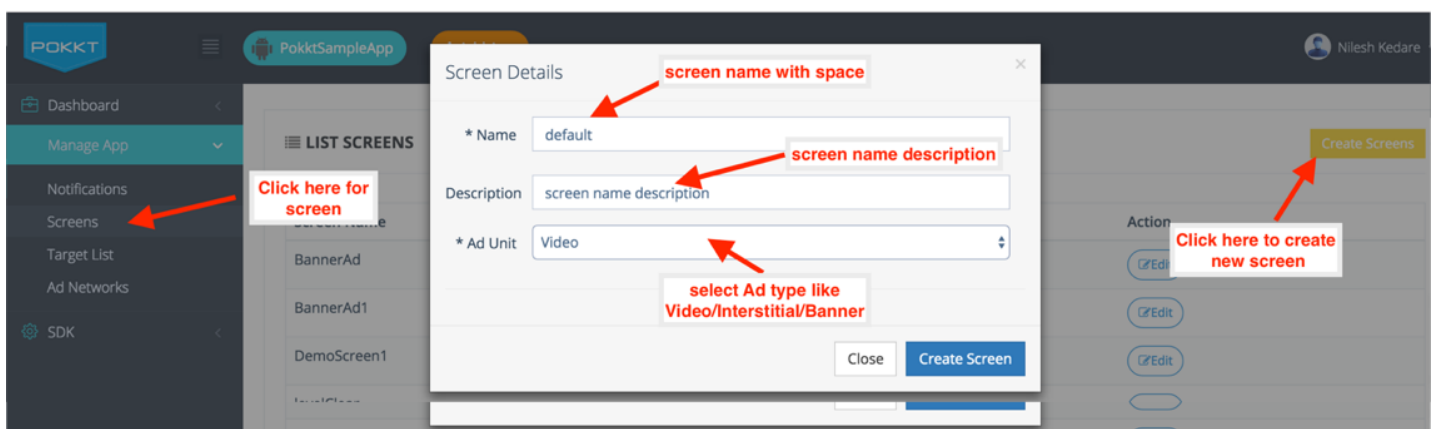
```
PokktManager.loadBannerWithRect(ScreenName, width, height, x, y);
```

6. **Banner Event:** Register event to get banner related callback like below:

```
//Register event
PokktManager.dispatcher.addListener(PokktEvent.BANNER_LOADED, onBannerLoaded);
PokktManager.dispatcher.addListener(PokktEvent.BANNER_LOAD_FAILED, onBannerLoadFailed);

// Handler method
protected function onBannerLoaded(event:PokktEvent):void {
    //event.screenName: same scree name for what banner ad request was done
}
protected function onBannerLoadFailed(event:PokktEvent):void {
    //event.screenName: same scree name for what banner ad request was done
    //event.message: Reason why banner load failed
}
```

7. **Screen Name:** For screen name, you will have to create in Pokkt dashboard. Please check below screen. This will help you to understand how to create new screen name in Pokkt dashboard.



Mediation Info

1. Pokkt SDK now supports 11 ad networks which you can integrate in your application for better monetisation.
2. To integrate these networks through Pokkt, please visit the mediation menu on downloads page and download AIR mediation zip and documentation zip files.
3. Please follow the mediation integration documents shipped for each network.
4. 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.
5. 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: *name, age, sex, mobileNo, emailAddress, location, birthday, maritalStatus, facebookId, twitterHandle, setEducation, setNationality, setEmployment and setMaturityRating.*

4. Important Points

Please do not copy the code points from this pdf as it may introduce unwanted characters and space in your code. Instead please refer to sample app source code in Pokkt bundle. Please also refer to sample app source code for better understanding of implementation.