



WP8 SDK v 1.0

Vserv.mobi WP8 SDK

Contents

Introduction.....	3
Prerequisites.....	3
Download the SDK	3
Integrating the Vserv SDK library.....	4
Requesting Vserv Ads	4
Intializing the AdController	4
Requesting Interstitials	4
Managed Interstitials using displayAd()	4
Requesting Banner Ads	5
Managed Banner Ads using RenderAd()	5
Essential Callbacks.....	5
No Fill Callback.....	5
Network Error Callback	5
Ad Close Callback	5
Queries	6

Introduction

The Vserv.mobi WP8 SDK is designed to help you integrate Vserv Ads in your application and fast track your way to monetizing your WP8 applications.

Prerequisites

Before you begin, we need to make sure of a few things:

1. You have a Vserv.mobi Developer Account. If you have one, go to Step 2. If you do not, sign up at our [Developer Portal](#).
2. Note down your **ZoneID**. This will be required later on when you use the WP8 SDK to request Ads. The default ZoneID will be available in your welcome email or you could visit <https://admin.vserv.mobi/admin/affiliate-zones.php> to get a list of all your zones.

Download the SDK

The first step to integrating Vserv SDK in your WP8 application is to download our SDK package. The SDK package is available from [here](#).

Alternatively the SDK can also be downloaded as NuGet package using the following command in the Nuget package manager.

```
Install-Package mobi.vserv.ads
```

We suggest that you unzip the SDK Zip file that you just downloaded into a temporary folder. Assuming that the folder is **[VSERV_SDK_FOLDER]**, you will see the following contents:

WP8V2SDK/	This folder contains the Vserv SDK
SampleApp/	Sample WP8 project source code that demonstrates Vserv SDK integration.

Integrating the Vserv SDK library

Before you start you need to make sure that your project satisfies the following requirements:

1. You need to copy the **vservWindowsPhone.dll** into the References folder of your project.
2. The **Images** folder should also be included in the project's root folder.
3. You also need to make sure that your project has the following capabilities:-
 - ID_CAP_NETWORKING"
 - ID_CAP_MEDIALIB_AUDIO"
 - ID_CAP_MEDIALIB_PLAYBACK"
 - ID_CAP_WEBBROWSERCOMPONENT"
 - ID_CAP_IDENTITY_DEVICE"
 - ID_CAP_IDENTITY_USER"
 - ID_CAP_PHONEDIALER
 - ID_CAP_REMOVABLE_STORAGE

You can then reference the Vserv library in your code by using file the using directive

```
using vservWindowsPhone;
```

Requesting Vserv Ads

The new SDK provides you with managed methods to request for Banners or Interstitials Ads in your WP8 application.

Intializing the AdController

Before you request for Ads you need to make sure you initialize the VservAdControl. Ideally you should initialize this when defining it in your project before the calling constructor.

```
VservAdControl VMB = VservAdControl.Instance;
```

Requesting Interstitials

Managed Interstitials using displayAd()

To request an interstitial you need to call the displayAd() method. Here is an example

```
VMB.DisplayAd(zoneID, LayoutRoot);
```

displayAd() is a part of the **VservAdControl** (*vservWindowsPhone library*) which need to be initialized before requesting for an Ad.

Optionally you can also set a timeout using the SetRequestTimeOut(timeout) method.

Please note: Please make sure you pass LayoutRoot, else the fullscreen ad will be displayed in the small banner.

Requesting Banner Ads

Managed Banner Ads using RenderAd()

You can request for Banner Ads which auto refresh automatically in a specific View using RenderAd() method. The signature for the method is shown below:

```
Grid grid = new Grid();  
grid.Height = 100;  
RenderAd(int ZoneID, grid);
```

Essential Callbacks

No Fill Callback

At times we might not be able to serve an Ad to you, in which case the Vserv SDK calls the following method. You would need to remove or hide the Banner Control when working with Banner Ads.

```
void VACallback_OnVservAdNoFill(object sender, EventArgs e)
```

Network Error Callback

At times when you request for an ad, there might not be any network available during the request, in which case the Vserv SDK calls the following method.

```
void VACallback_OnVservAdNetworkError(object sender, EventArgs e)
```

Ad Close Callback

This is an essential callback which you need to implement. The Vserv Ads make use of the ApplicationBar to provide a native experience to the user, this is achieved by taking over your ApplicationBar. Thus, ideally you need to reinitialize your ApplicationBar in this callback. This is called when the user returns to your App from the Ad

```
void VACallback_OnVservAdClosing(object sender, EventArgs e)
```

Queries

If you have any Queries or issues please email us at developers@vserv.mobi