

Version

1

ENCORE SYSTEMS LLC

---

Web Development and eCommerce Integration

# Encore PayPal SDK Developer Guide

# Encore PayPal SDK Developer Guide

---

The Encore Systems PayPal Software Development Kit supersedes the Encore PayPal Class Libraries. More than a wrapper for the Website Payments Pro API, the SDK provides many new features. One of the most important features is that it is a unified SDK for both Website Payments Pro and the new Adaptive Payments API. The most exciting new feature is the drag-and-drop PayPal controls. With absolutely no API programming, the developer can drop a control on the design surface, set a few properties, and have that particular API transaction ready to go.

The SDK no longer distinguishes between SOAP, NVP, or XML forms of the APIs – it just provides a clean, object oriented interface designed to simplify PayPal integration with ASP.NET websites and .NET applications. The Encore SDK has a short learning curve and is well documented so that developers can spend their time designing their application and know that the PayPal integration will just work.

# Table of Contents

Overview .....	2
Features .....	2
FAQ .....	3
Quick Start.....	5
Templates.....	5
Videos .....	6
How to Use the SDK .....	7
Encore SDK Reference .....	8
API Classes.....	8
API Object Container Classes.....	11
Special Classes, Constants, and Binding Sources .....	12
PayPal API Analysis Application .....	13

## Overview

**W**ebbsite Payments Pro has been around for a few years now, and it is a great improvement over Standard Website Payments. It handles all of the basic payment transactions very well, is feature-rich, but fairly inflexible. PayPal has addressed this issue by opening up their platform to developers. The first step in this process is the release of Adaptive Payments which allows much more flexibility in defining how money is transferred between buyers and sellers.

Adaptive Payments allows scenarios such as having a single payment split among multiple receivers, in the case of selling items from multiple vendors in a single shopping cart, for instance. Another new feature is allowing “chained” payments in which the buyer pays a single receiver, but that receiver can pass part or all of the payment on to multiple secondary receivers. In addition, the merchant can designate which parties incur the payment fees.

## Features

- The SDK is designed to run under medium trust. This allows use in hosted environments such as GoDaddy, HostMySite, 1and1, RackSpace, etc.
- All often-used properties have built-in defaults that make sense in most situations. These default property values can be overridden globally in the application’s configuration file. Every default can be overridden on a per-call basis by setting the property directly on the API call object.
- All authentication credential properties such as Username, Password, Signature, etc. can be globally defaulted in the application’s configuration file. The SDK includes a built-in encryption function to make these settings more secure. For increased security, pass your own credentials function to the credentials delegate and the SDK will call your function as necessary.
- Complete Visual Studio Intellisense on all SDK members. Each member shows requirements, limitations, possible values, etc. Numeric values are stored as decimal and integer types, instead of string type. Arrays are represented by `Collection<T>` to simplify manipulation.
- All class members are serializable for ability to be stored in view state, session state, profiles, etc. Enumerations, such as country codes, currency codes, state codes, etc., commonly used in the user interface have a `BindingSource` property to allow binding to an `ObjectDataSource` or directly to `DropDownLists`, etc.

- Includes a complete diagnostic tool with documented source code showing every possible API call. Try any possible combination of property settings on API calls to see what PayPal's response is. Cut and paste the examples to get a quick start on your application.
- New Features:
  - New drag-and-drop controls – no API coding necessary! Drop a control on your design page, set the required properties (either statically or dynamically,) and handle the OnSuccess and OnFailure events.
  - New Adaptive Payments API provides parallel payments allowing single payments to be split among multiple recipients. Use this to have a shopping cart with multiple items from different vendors.
  - New Adaptive Payments API provides chained payments allowing a primary recipient to automatically and invisibly pass portions of the payment along to secondary recipients. Perfect for collecting commissions.
  - All code completely rewritten following Microsoft's .NET Framework Design Guidelines – verified against FxCop and StyleCop.
  - Supports all Website Payments Pro API calls as of Version 58.0.
  - Supports all Adaptive Payments API calls as of Version 1.1.0.

## FAQ

Q. Will the SDK still work if PayPal updates their API?

A. Yes. The SDK was developed for a specific version of the API. As long as that version is used, the SDK will continue to work as designed. Newer versions of the API do not break the older versions.

Q. What is the GetCredentials function in the example website?

A. This is an example of using the credentials delegate. The GetCredentials function returns an SdkCredentials class. Any function returning this class may be used as a delegate for setting the API credentials in each API call.

Q. Should I use the GetCredentials function?

A. No. The GetCredentials function is designed for the online demo site. It stores the API credentials in Session so that multiple developers can use the site to test the SDK. This would not be a good solution for a production site. Either store the credentials in the web.config or create your own delegate function that gets the credentials from a database, for example.

Q. Can I just set the API credentials in the code that sets the rest of the Request properties?

A. Yes. But, that would require setting them on every API call. Using the configuration file or a delegate is more efficient.

Q. Why am I getting Error 10501 Invalid Merchant Configuration?

A. This means that Website Payments Pro is not enabled for your account. You need to accept the billing agreement, or contact PayPal customer service to have your account configured correctly.

## Quick Start

Visual Studio templates are provided for getting started quickly on various PayPal integration projects. The SDK installation package is a Visual Studio Content Installer file. Just select the appropriate Project template and you will have with a completely configured, running application to give you a quick start. Or, select the appropriate Item template to add the SDK to your existing project. The following templates are included:

### Templates

- Example Site (C#) An ASP.NET web application demonstrating every possible API call. Use this C# application to explore the Encore PayPal SDK. Every API call can be built interactively from a form providing every possible parameter for that call. The code behind file completely documents how each API object is built, configured, and called.
- Example Site (VB) Same as above for Visual Basic.
- Web Site (C#) A single page ASP.NET web site with the SDK installed and configured. Use this C# application to start building a PayPal integrated site from scratch, or to see the minimum settings necessary to start one.
- Web Site (VB) Same as above for Visual Basic.
- Web Application Item (C#) Use this C# item template to integrate PayPal into an existing web application.
- Web Application Item (VB) Same as above for Visual Basic.
- Winform Application (C#) A Windows Forms application with the SDK installed and configured. Use this C# application to start building a PayPal integrated desktop application from scratch, or to see the minimum settings necessary to start one.
- Winform Application (VB) Same as above for Visual Basic.
- Winform Application Item (C#) Use this C# item template to integrate PayPal into an existing desktop application.
- Winform Application Item (VB) Same as above for Visual Basic.

## Videos

Check out these short videos for a quick start on various tasks.

[SDK Installation](#)

[Obtain and Install License Keys](#)

[Create the Example Application](#)

[Using the Example Application](#)

[Add the SDK to an Existing Project](#)

[Drag-and-Drop Controls Adaptive Payments](#)

[Drag-and-Drop Controls Website Payments Pro](#)

## How to Use the SDK

**T**he Encore SDK follows the structure of PayPal's API as closely as possible, but greatly simplifies it by creating all of the necessary objects automatically when the developer sets the desired properties of the main API call object. The naming convention of the API call object properties follows the original API, so that PayPal's API documentation can be used to find the desired request and response properties. In addition, all of the properties have complete Intellisense documentation so that the developer can see all relevant information of each property without referring to the API documentation.

The SDK provides a class for each API call. The three main members of these classes are the Request member, the Response member, and the Post() member. To complete an API call, the developer creates an instance of the desired class, sets the required properties of the Request member, calls the Post() member, then reads the results from the properties in the Response member.

## Encore SDK Reference

Follow this link to get the [latest documentation from PayPal](#). The following SDK API classes correspond directly to the API calls listed in the *Adaptive Payments Guide* and the *SOAP API Developer Reference*. All classes are prefixed with “Sdk” to differentiate them from the actual API calls. This allows access to the SDK and the actual API without fully qualifying each member.

### API Classes

- SdkAddressVerify returns the status of a PayPal account address, including street and zip code match results. (requires special permission)
- SdkBillingAgreementUpdate updates or deletes a billing agreement.
- SdkBillOutstandingAmount bills any balance due on a recurring payments profile.
- SdkCreateBillingAgreement returns a billing agreement ID that may be used to make reference transactions against a PayPal account. (undocumented, needs special permission)
- SdkCreateRecurringPaymentsProfile creates a recurring payments profile allowing automatic periodic billing.
- SdkDoAuthorization is your request to authorize a customer order that can be fulfilled within 29 days.
- SdkDoCapture is your request to completely or partially settle an order, an authorization, or reauthorization.
- SdkDoDirectPayment charges or authorizes a credit card.
- SdkDoExpressCheckoutPayment obtains payment through Express Checkout for a final sale or requests authorization for later capture of payment.
- SdkDoNonReferencedCredit makes a non-referenced credit to a customer’s credit card.
- SdkDoReauthorization reauthorizes a previously authorized a transaction and that has passed its settlement period.

- SdkDoReferenceTransaction makes a payment based on the information provided in a previous Direct Payment, avoiding the need to provide any buyer information.
- SdkDoVoid voids an order or an authorization.
- SdkGetBalance returns the PayPal account balance of the account whose API credentials are used for the call. (undocumented)
- SdkGetBillingAgreementCustomerDetails returns information about the customer, including name and address on file with PayPal.
- SdkGetExpressCheckoutDetails returns information about the customer, including name and address on file with PayPal.
- SdkGetRecurringPaymentsProfileDetails returns the details of a recurring payments profile.
- SdkGetTransactionDetails gets details about a transaction.
- SdkManagePendingTransactionStatus accepts or denies a pending transaction. (undocumented, untested, needs special permission)
- SdkManageRecurringPaymentsProfileStatus allows a recurring payment profile to be cancelled, suspended, or reactivated.
- SdkMassPay processes payments en masse to up to 250 different recipients.
- SdkPay Use the **Pay** API operation to transfer funds from a sender's PayPal account to one or more receivers' PayPal accounts. You can use the **Pay** API operation to make simple payments, chained payments, or parallel payments; these payments can be explicitly approved, preapproved, or implicitly approved.
- SdkPaymentDetails Use the **PaymentDetails** API operation to obtain information about a payment. You can identify the payment by your tracking ID, the PayPal transaction ID in an IPN message, or the pay key associated with the payment
- SdkPreapproval Use the **Preapproval** API operation to set up an agreement between yourself and a sender for making payments on the sender's behalf.
- SdkPreapprovalDetails Use the **PreapprovalDetails** API operation to obtain information about an agreement between you and a sender for making payments on the sender's behalf.
- SdkRefund The **Refund** API operation refunds all or part of a payment.
- SdkRefundTransaction refunds a payment.
- SdkSetCustomerBillingAgreement indicates to PayPal that you are using Express Checkout to create billing agreements for your customer.
- SdkSetExpressCheckout indicates to PayPal that you are using Express Checkout to obtain payment from your customer.
- SdkTransactionSearch searches transaction history.

- `SdkUpdateRecurringPaymentsProfile` allows modification of the terms of a recurring payments profile.

## API Object Container Classes

Container classes such as Request, Response, Credentials, Address, Payment, CreditCard, etc. need not be instantiated when creating an API call, but they may be used to create objects that can be used separately from the API calls, stored in view state, session state, profiles, etc. These classes are also prefixed with “Sdk” to differentiate them from the API members.

## Special Classes, Constants, and Binding Sources

- `SdkCredentialsDelegate` The developer can define a function that takes no parameters and has return type `SdkCredentials`, and then pass that function to any API call instead of setting credentials individually. This can increase security by passing the API credentials from an encrypted store of the developer's choice. The delegate function may also be declared in the application's configuration file to make this a global setting.
- `SdkConfig` is a static class that returns all properties defined in the `encore.PayPal` configuration section of the application's configuration file. Default values are returned for any property not set in the configuration file.
- `SdkGetTransactionDetailsResponse.BindingSource` returns a single transaction details record in the form of a `Collection<T>` that allows binding directly as the data source for a `FormView` or `DetailsView` object.
- `SdkMonthCodeType` encapsulates a list of calendar month constants, including integer value, short name, and long name. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `SdkYearCodeType` encapsulates a list of calendar year constants from the current year through current year plus eleven. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `SdkStateCodeType` encapsulates a list of states and provinces, including name and two-character code. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `SdkCountryCodeType` encapsulates a list of countries, including name and two-character code. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `SdkLocaleCodeType` encapsulates a list of countries for which PayPal Express Checkout pages include culture settings. Includes name and two-character code, a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `SdkCreditCardTypeType` encapsulates a list of credit card types accepted by PayPal. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `SdkCurrencyCodeType` encapsulates a list of currencies accepted by PayPal, including name and three-character code. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- The `SdkTraceExtension` class captures the XML envelopes that comprise the raw API request and response data. This allows the developer to see exactly what data is sent to and received from the API – invaluable for troubleshooting.

## PayPal API Analysis Application

Included with the Encore PayPal SDK is a web application that serves many important purposes. This application demonstrates every PayPal API call. The code-behind files for each API call are well documented and show all required and optional parameters. Modular user controls for collecting and presenting PayPal API call parameters are readily available for possible adaption in your own application.

In addition, the example application serves as an analysis tool for the PayPal API. Every parameter can be set interactively when making API calls. This allows the developer to test every possible combination of parameter settings to see what PayPal's response will be.

Enter your live API credentials and the example application becomes an interactive PayPal terminal. Use it to research transactions, capture authorized payments, make refunds, modify recurring payments profiles, check your balance, enter payments for phone orders, etc.