# Cardsave Payment Gateway

## Cart Implementation

## Specification and Help

**David McCann**
**Cardsave Online**

**Version 1**

**1st August 2010**

## Contents

# Overview

The Cardsave Payment Gateway offers a secure, reliable platform for merchants to take payments online. This document provides the specification and help in order for a developer to integrate the Cardsave Payment Gateway into an online shopping cart. This includes access to generic code application packs for various language and detailed documentation on the gateway API.

CardSave is authorised by a number of banks to offer competitive acquiring rates under a Buy Group type relationship. We group many merchants volume together allowing individual Merchants to qualify for rates they would not be able to achieve on their own.

The Cardsave solution has a comprehensive Merchant Management System that includes a virtual terminal and transaction reporting. The merchant management system shows off functionality available through the API, with features like recurring billing, scheduled transactions and pay-by-link.

## Integration Types

There are four possible types of integration into the Cardsave gateway.

- Direct/Integrated
- Redirect/Hosted
- Transparent Redirect
- And Pay-By-Link.

The Cart implementation should include the direct/integrated, redirect/hosted and transparent redirect options to give the most flexibility for implementation. Pay-By-Link is only generally used through the Merchant Management System by the merchant to collect specific payments or the link generated can be used for applications such as a simple donation page.

## Direct/Integrated (Preferred Method)

Card details are taken directly on the site. An SSL is required. This provides the most flexible technical approach to communicating with the gateway. This provides a superior user experience for the customer.

## Re-direct/Hosted

The customer is passed to the Cardsave secure server in order to take the payment. Messages are sent back to the merchant's website regarding the status of the transaction. The hosted page can be skinned but the customer is taken away from the merchant's website. Developers often choose this option because they feel more familiar with the integration method, no SSL is required and security on shared servers is less of an issue.

## Transparent Redirect

To the customer, taking payment on the site using this method is no different to the direct/integrated solution. The transparent redirect offers an integration solution that is halfway between the direct integration and the hosted payment form. Using it allows merchants to appear to host their own payment pages, without the card details ever touching their systems. An SSL is still required.

## Account Types within the Gateway

There are several types of merchant accounts. ECOMM – Used by the website to communicate with the gateway. MOTO – Mail Order/Telephone Orders used by the Virtual Terminal. There are also variations for continuous authority and for special cards like AMEX. For the purpose of creating a payment gateway for a standard shopping cart you only need use the ECOMM type.

## Test Gateway Accounts

Please see "**Registering an Account**", which is "Step 1" on page 5. This will give you access to the Merchant Management System and a test gateway account. Before a merchant account has been approved by Cardsave the merchant is sent login details for their Merchant Management System and a test gateway account so development can start work. Often these details are sent to the developer via the merchant instead of the developer registering a new account. The "**Ecomm Test Gateway**" login and password details should be used by the website to communicate with the gateway. Please do not confuse this with the Merchant Management Login details. The MerchantID used by the gateway is different to the merchant number provided by acquiring bank.

## 3D Secure

The 3D-Secure system is a scheme implemented by the card schemes (primarily Visa, who call it "Verified by Visa" or "VbV" and "MasterCard", who call it "MasterCard SecureCode"). This is a requirement of the payment gateway implementation and will affect the payment process flow for the direct/integrated and transparent redirect payment page methods.

When a payment is processed and authorised through the 3D secure model liability for the payment shifts from the merchant, to the acquiring bank for all fraudulent transactions. For more details on how to implement this please see the section "**3D Secure - Coding**" on page 17.

## Testing and Making it live!

The Cardsave gateway has the same functionality in test mode as in the live environment. The difference between the live and test environment being the MerchantID, Password and for the hosted and transparent solution the pre-shared key. Cardsave provide test cards which produce various results that can be used to test the gateway. Please see "**Testing the Gateway Implementation**" on page 18 for more details.

## First Steps

I would suggest the first step is to try out the gateway using the **"Quick Start"** on page 5. In doing so this will ensure the environment you are working with is setup as required.

# Quick Start – Try the Cardsave Gateway

Try the gateway for yourself in your development environment? Follow the three steps below in order to test the payment gateway using the generic code examples.

## STEP 1 – Register, (If you haven't already done so)

If you haven't already created a test account setup please use the link below to register...
***https://mms.cardsaveonlinepayments.com/Pages/PublicPages/RegisterMerchant.aspx***

- You will then receive a verification email to which you must respond to complete the registration process.
- You will then receive two separate emails with the login details to the back office management system and payment gateway.
- When you login to the Merchant Management System for the first time, you will be asked to change your password.

## STEP 2 - Get the Example Code – Direct/Integrated Method

Please read the "**Direct/Integrated Payment Page Solution**" requirements on page 12 before proceeding. Then download the integration pack.

***https://mms.cardsaveonlinepayments.com/SiteFiles/VirtualFiles/GATEWAY_INTEGRATION_PACK/IntegrationPack.zip***

Read the readme file for your preferred language. Copy the code to a test area on your site and change the configuration file as required. Login details, baseURL and payment processor domain (cardsaveonlinepayments.com) will require configuring.

There is also example code for the "**Redirected/Hosted Payment Page Solution**" on page 13.

## STEP 3 – Test and View Your Transactions in the Merchant Management System

You will now be able to test the payment page by submitting test transaction and viewing them in the back office Merchant Management System under the "Test Account", "Transaction History" in the menu. You can download the test cards here…

***https://mms.cardsaveonlinepayments.com/SiteFiles/VirtualFiles/TEST_CARD_DETAILS/TestCardDetails.zip***

## Building the Cart – Gateway Administration Page

The Cardsave payment page configuration section is where the merchant/developer configures the payment module. The variables determine the behaviour between the cart and the payment gateway, the link between the two.

In the administration section include a direct link to the Cardsave MMS for the convenience of the merchant, ***https://mms.cardsaveonlinepayments.com***. Please also include a link to the Cardsave developer's website page. Developers can find help here including example code, the latest cart plug-ins and graphics, ***http://www.cardave.net/support***.

Any environment or language requirements not fulfilled on the merchant's server for a particular integration method should be displayed as a warning in the Gateway administration page. E.g. If PHP-cURL is required by the gateway code and it is not enabled on the server then the developer/merchant should be warned so they can install it. Please see the requirements section for each integration type but generally this is only an issue with direct/integrated payment method.

Below are the variables to be configured. **"General Settings"** are required by one or more of the integration methods. If the user changes the "Payment Integration Method" to "Re-direct/Hosted" or "Transparent Redirect" additional required variables should be asked for. See the appropriate section. Please note all variables are not applicable for every cart implementation and some variables should not be editable but could be shown in the setup, please see the editable column.

### General Settings

| Variable | Default | Description | Editable |
|---|---|---|---|
| Title | Cardsave | Title to display on the customer payment form that describes the gateway | Yes |
| Enabled | Off | Gateway On/Off | Yes |
| Debug Mode | No | Show debug information when posting to the gateway | Yes |
| New Order Status | Processing (Dependent on the cart) | Status of the order before processing | Yes |
| Quick Test Mode | Off | On/Off – If enabled a dropdown with test card details are displayed on the payment page. When selected the Card Details are pre-populated to save testing time. | Yes |
| MerchantID | | Gateway Merchant ID | Yes |
| Password | | Gateway Password | Yes |
| Payment Processor Domain | Cardsaveonlinepayments.com | The URL of the Gateway used in the Integrated Solution | Yes |
| Payment Processor Port | 4430 | Port used when communicating out to the gateway. Integrated solution only. | Yes |
| Payment Action | Authorise and Capture | Type of transaction to be performed. "Authorise and Capture"-"SALE" or just "Authorise"-"PREAUTH". To collect the money the merchant will have to "COLLECT" the payment within the MMS System. | Yes |
| Payment | Integrated Direct(API) | Either Direct/Integrated, Redirect/Hosted or | Yes |

| | | | |
|---|---|---|---|
| Integration Method | | Transparent Redirect | |
| Pre-Shared Key (You can find this in your MMS under Merchant Details) | | Pre-Shared Key used by hosted and transparent redirect integration methods. This can be changed or viewed in the merchant details in the MMS. | Yes |
| Hash Method (You can find this in your MMS under Merchant Details) | SHA1 | Choices are SHA1, MD5, HMACMD5, HMACSHA1. This can be changed or viewed in the merchant details in the MMS. | Yes |
| CV2 Mandatory | True | True/False. Make CV2 compulsory | Yes |
| Address 1 Mandatory | True | True/False. Make Address 1 compulsory | Yes |
| City Mandatory | True | True/False. Make City compulsory | Yes |
| Postcode Mandatory | True | True/False. Make postcode compulsory | Yes |
| State/County Mandatory | True | True/False. Make state compulsory | Yes |
| Country Mandatory | True | True/False. Make Country compulsory | Yes |

### For RE-DIRECTED/HOSTED Integration Method display the additional variables

| Variable | **Default** | Description | Editable |
|---|---|---|---|
| Hosted Payment Form URL | ***https://mms.cardsaveonlinepayments.com/Pages/PublicPages/PaymentForm.aspx*** | URL of the hosted Payment Form | No |
| Result Delivery Method | Server | Server/Post. Server is the preferred method as this sends the results back to the server directly rather than through the customer's browser. Using the post method can result in the merchant not being notified of an order due to browser security. | Yes |
| Server Result URL | | The merchant's external server URL used for SERVER result delivery method. | Yes |
| Display Results Page on Hosted Form | | Boolean that determines whether the payment result will be displayed on the hosted payment page, or redirected to the merchant's site after a response from the merchant's external server. | Yes |

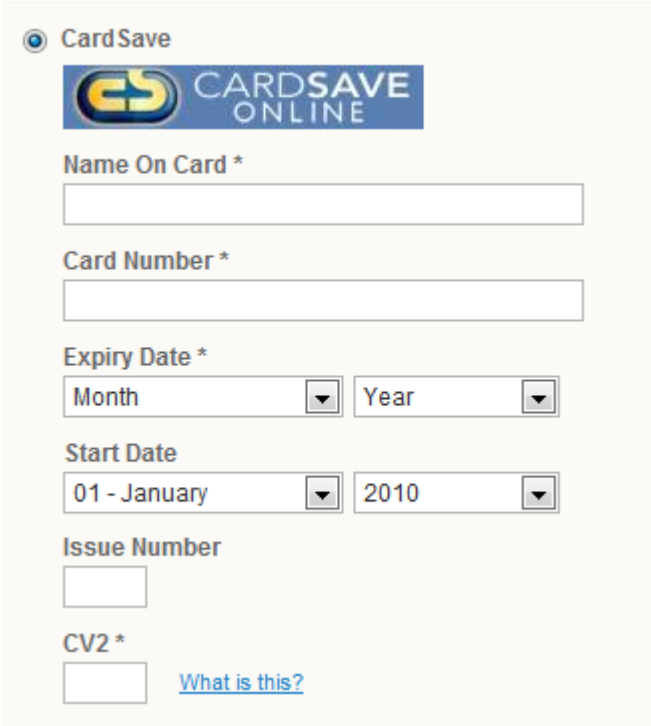
### For "TRANSPARENT REDIRECT" Method, display the additional variables

| Variable | Default | Description | Editable |
|---|---|---|---|
| **Transparent Redirect Action URL** | ***https://mms.cardsaveonlinepayments.com/Pages/PublicPages/TransparentRedirect.aspx*** | URL of the Transparent Redirect where actions are to be posted | No |

## Building the Cart – Customer Card Entry Page

This is the page were the customer enters their card details. If the redirect/hosted payment page method is selected by the merchant then the cart should show the Cardsave payment page logo and the message, "You will be re-directed to our external secure payment page to take the payment". When the customer continues they are redirected to the Cardsave server to take the payment. Once payment is processed on the redirected/hosted payment page, the status of the payment is returned, please see the "**Re-directed/Hosted Payment Page Solution**" section in this document to see how the "Server" post back method can be implemented.

For the direct/integrated and transparent redirect methods the card details are collected on the website page. This is an example of the Magento Cardsave plug-in. Remember you should store none of the card details on the merchants server.




For direct/integrated and transparent redirect payment gateway method are used and the cart setting "Quick Test Mode" is set to "True" then an additional dropdown box should be offered populated with the descriptions of default test cards. Once selected the test card details should populate the fields on the payment page automatically, this makes the gateway far easier and quicker to test for the developer/merchant.

It is essential to understand the cart and cart API when you are developing the plug-in. You should consider if the payment page needs to ask for the address details of the card holder. Is it ok to send the goods to a different address other than that of the payee?

Display the Cardsave payment page logo as this adds an element of trust to the payment page. The variables set in the payment page configuration should be taken into consideration when validating the details. E.g. Is the CV2 or address line 1 a required entry.

Next to the CV2 either display a "What is this?" logo or include a "What is this?" link to a CV2 explanation page. Please use the appropriate validation and use dropdown calendars where

appropriate, although validating the card number before hitting the gateway is not essential. There should be appropriate validation checks make before any attempt is made to the gateway. The direct integration document API documents the variable types and the maximum variable sizes, please see the direct/integrated API document.

***https://mms.cardsaveonlinepayments.com/SiteFiles/VirtualFiles/GATEWAY_INTEGRATION_DOCS/IntegrationDocuments.zip***

The payment processor will return a payment status code after the payment is processed indicating the state of the payment. Please take care in re-directing the customer to the appropriate page and show the appropriate message. The payment authorisation code and order ID should be displayed to the customer. Appropriate emails e.g. order confirmation, invoice should then be sent to the customer and merchant for authorised transactions.

Caution should be used in directing the customer dependant on the status response. e.g. if the gateway responds with invalid card details, CV2 failed, Address check failed or post code check failed then the process flow should allow the customer to retry without having to enter all their details again. Where appropriate position the customer on the correct field that needs correcting on the payment page. Any gateway communication errors such as “Cannot communicate with payment gateway” should show user friendly error messages instead of technical ones, e.g. “We cannot take your payment right now please contact us to place your order”.

# Implementation and Coding

The implementation of the payment gateway into the cart is to allow the developer/merchant to use any one of three methods of integration. The implementation of the three methods is described in more detail in the next three sections of this document.

## Best Coding Practices

Good programming practices and an awareness of shared server issues should be taken into account. Some hosting providers now provide tightened frameworks and limitations for security on shared servers. An issue we've had with a gateway implementation recently is the cart passing long URL's as variables to the payment pages through the address bar. The hosting servers did not allow any variable beyond 430 characters which produced error in the payment page integration. Good implementation now will save the developer and merchant time is trying to resolve similar issues.

Code is to be commented throughout. Full code version control and documentation is to be implemented.

## Cart Functionality

The implemented code is to work as one with the carts functionality. Currency and language API's are to be implemented fully it is the requirement of the implementation to translate this when communicating with the payment gateway.

## Hashing

For redirected/hosted and transparent redirect payment page solutions either one of 4 hashing methods can be used; SHA1 (default), HMACMD5, MD5 and HMACSHA1. Ideally all these methods should be available to the merchant/developer to use. The MMS allows the developer/merchant to configure the receiving hash method and key on the gateway servers. There can be slight variations on which variables are used and how the hash is constructed depending on the hashing method chosen. Please see the detail in the redirected/hosted API documentation in the "**Redirect/Hosted Payment Page Solution**" section on page 13.

# Direct/Integrated Payment Page Solution

In this method, card details are taken directly on the site. This provides the most flexible technical approach to communicating with the gateway and provides a superior user experience for the customer. We provide fully working generic code examples within the integrated code pack.

## Requirements

The live environment requires an SSL although but it can implement and tested without a SSL. The gateway requires port 4430 on the merchant's server to be open for outgoing TCP. The example PHP code implementation will work in PHP 5.1.6 or above. SimpleXML and cURL are also required for PHP. Classic ASP implementation requires a DLL to be installed in the Windows server system folder. For more details about the requirements for specific languages please read the readme file in the code pack.

## Code Examples

Below is the URL to the latest code pack. There are working examples for various languages. Please read the readme file in each language section for more information about installing and how to use them.

***https://mms.cardsaveonlinepayments.com/SiteFiles/VirtualFiles/GATEWAY_INTEGRATION_PACK/IntegrationPack.zip***

## API Documentation

This document provides detailed information about the Cardsave Gateway API.

***https://mms.cardsaveonlinepayments.com/SiteFiles/VirtualFiles/GATEWAY_INTEGRATION_DOCS/IntegrationDocuments.zip***

## Polling Servers

You will see in the code examples that this code implements three gateway entry points. Cardsave have three servers in three different locations to allow for disruption of service either at the ISP where the server is housed or allowing for a complete server failure. Your code should implement the model in the example code polling all three servers if necessary. If server gw1 fails to respond server gw2 should then be poled and if that fails gw3. This fall over architecture makes the gateway reliable but a little more work is required to implement this within the cart plug-in as the code needs to reside on the merchant's server.

# Re-direct/Hosted Payment Page Solution

Using this method the customer is passed to the Cardsave secure server in order to take the payment. Messages are sent back to the merchant's website regarding the status of the transaction. The hosted page can be skinned but the customer is taken away from the merchant's website. Developers often choose this option because they feel more familiar with this integration method, no SSL is required and security on shared servers is less of an issue.

## Requirements

There is no need to use an SSL and a basic installation of the base language is all that should be required in order to implement this method.

## Code Examples

Here are basic code examples for PHP, Classic PHP and ASP.NET. These examples do not implement the "Server" postback method please read further on in this chapter for instruction on how to implement this.

PHP – ***www.cardsave.net/downloads/PHPHostedPaymentFormAPR2010.zip***

Classic ASP - ***www.cardsave.net/downloads/ASPHostedPaymentFormAPR2010.zip***

ASP.NET – ***www.cardsave.net/downloads/DOTNETHostedPaymentFormMAY2010.zip***

## API Documentation

The detailed API for the hosted page is below.
https://mms.cardsaveonlinepayments.com/Pages/PublicPages/PaymentForm.aspx

This URL above in its raw form is the redirected/hosted gateway helper/instruction page which gives instruction on how to construct the http post for the redirect payment method. This also provides documentation on the returning POST variables. The payment gateway will 'Post' back to the URL specified in your Initial post. You can then read the returning data to determine if the transaction has passed or failed and if so why. The "Status", (now available to you) can then be used to control your action on the website.

## Post URL

***https://mms.cardsaveonlinepayments.com/Pages/PublicPages/PaymentForm.aspx***

## Hash Help Tool

***https://mms.cardsaveonlinepayments.com/Pages/PublicPages/PaymentFormHelper.aspx?HelperType=PaymentForm***

## Server Postback Method

Some security updates in various browsers now interrupt the postback to the customer's browser. It is possible that the merchant's server is not informed if a payment has been taken by the payment gateway. To combat this we have implemented a server postback method for the result of the transaction directly to the merchant server and not via his browser.

The process is…

1) The website posts to the Cardsave hosted payment form with a unique OrderID.
2) The card details are entered on the hosted payment form and submitted to the payment gateway, a result is returned to the Cardsave hosted payment form.
3) In the case of "ResultDeliveryMethod=SERVER" the results are posted directly to the merchants server, (as directly to the server no hashing is required, no need to check).
4) The website server (from the URL it was posted to), sends back a response to confirm delivery as we cannot guarantee delivery of the result.
   StatusCode=XX&Message=xxxxxx
   **There should not be ANY other characters in the response (including white space, or any HTML). Here are some valid examples:**
   StatusCode=0
   StatusCode=0&Message=Results received OK
   StatusCode=30&Message=Database timeout error
   StatusCode=30&Message=Unhandled exception
5) Once the confirmation message is received by the payment gateway for "ResultDeliveryMethod=SERVER" and "PaymentFormDisplaysResult=false" a further message is posted back to the "callbackURL". This message includes the CrossReference and OrderID for reconciliation of the response.
6) The customer is also returned to the website "callbackURL".

Further details can be found in the hosted payment form API.

## Skinning a Hosted Payment Form

It is not a requirement of the cart implementation to touch the skinning of the hosted payment form but for the developer's curiosity please see the "readme" document within the skin pack ***http://www.cardsave.net/downloads/HostedPaymentFormSkinning.zip***. This explains the skinning process and gives examples. Once the skin is created the developer simply emails ***ecomm@cardsave.net*** and Cardsave will implement it for the specified merchant.

# Transparent Redirect Payment Method

To the customer, taking a payment on the site using this method behaves much like the direct/integrated solution. The transparent redirect offers an integration solution that is halfway between the direct integration and the hosted payment form. Using it allows merchants to appear to host their own payment pages, without the card details ever touching their systems.

## Requirements

An SSL is required but otherwise the requirements are the same as the re-direct/hosted payment page solution.

## API Documentation

***https://mms.cardsaveonlinepayments.com/Pages/PublicPages/TransparentRedirect.aspx***

## Post URL

***https://mms.cardsaveonlinepayments.com/Pages/PublicPages/TransparentRedirect.aspx***

## Hashing Helper Tool

***https://mms.cardsaveonlinepayments.com/Pages/PublicPages/PaymentFormHelper.aspx?HelperType=TransparentRedirect***

## Hashing and the Pre-Shared Key

The re-direct/hosted and transparent redirect solutions implement a hashing algorithm. This is implemented by the gateway to make sure the variables passed to the gateway have not been tampered with in transit. All the variables are stringed together along with the password and pre-shared key, (which are not passed in transit) and a hash string is produced. When the “posted” is received by the gateway the hash is reconstructed and compared to see if it matches the one sent in the post. If they are different then an error occurs and the process is stopped.

We have constructed a program to help you test your hash digest output. Please see the appropriate integration method for the URL of the helper tool and ensure the hashing is built correctly as documented for the appropriate payment method and post method.

### The Pre-Shared Key

The pre-shared key is unique to each merchant. This can be viewed/changed within the Merchant Management System. Login and select merchant information from the left menu. At the bottom of that page you will see the security information and pre-shared key.

## 3D Secure - Coding

A full and detailed description of how to implement 3D secure is contained within ***appendix 5 of the direct integration guide***. This can be downloaded here…

***https://mms.cardsaveonlinepayments.com/SiteFiles/VirtualFiles/GATEWAY_INTEGRATION_DOCS/IntegrationDocuments.zip***

The code examples in the integration pack implement this model. Please see the "**Direct/Integrated Payment Page Solution**" on page 12.

## Testing the Gateway Implementation

When testing your implementation of the Cardsave Payment Plug-in please take the following into consideration, (I would suggest you start by testing on a server with the minimum installed.)

1) If the merchant is using the re-directed/hosted solution and the direct/integrated solution has not got the correct installed components for a direct/integrated solution does the hosted method still work? None of the payment methods should be dependant on another method of integration.
2) Using the test cards scenarios do the payment pages behave correctly for all methods of integration? Does the user experience work as expected?
3) Are the order and invoice status within the cart updated correctly?
4) Are appropriate emails sent to the merchant and customer? This includes the order confirmation, invoice and payment notifications.
5) While testing are all the URL's in the address bar of an appropriate length and are any of the variables passed through the address bar too long?

## About Cardsave

Cardsave was formed in 1995 by an independent retailer frustrated by the high costs of credit and debit card processing available for his business, compared to those offered to the large multiples. In order to negotiate lower rates he needed to drastically increase his buying power, and this meant joining forces with others in the same position.

This was the birth of Cardsave, a unique buying group dedicated to providing competitive business services to the independent retailer. Our reputation for saving money grew quickly and today we have more than 39,000 members.

In addition to low cost credit card processing, we offer a wide range of additional good value business services to all our members.