



Merchant Integration Guide
Hosted Pay Page v. 1.2.5



Revision Number	Date	Description
V1.1.0	January 2, 2007	Document edited for coherence
V1.2.0	July 23, 2008	<ul style="list-style-type: none"> -Section 5. MasterCard SecureCode <ul style="list-style-type: none"> -Added MasterCard SecureCode. -Section 3. Skills and System Requirement <ul style="list-style-type: none"> -Added PCI & PA-DSS compliance note. -Section 7-D. Shipping and Billing Information <ul style="list-style-type: none"> -Added sample code. -Section 7-E. Added Optional 'rvar' Variables. -Section 12. Performing a Transaction Verification <ul style="list-style-type: none"> -Added Transaction Verification Error codes -Added Sample Code -Section 13. Gift Cards <ul style="list-style-type: none"> -Added Gift Card transactions -Section 15. Special Error Codes <ul style="list-style-type: none"> -Added Special Error Codes and Message/Description. -Added Appendix E. XML POST Response for Financial Transaction -Added Appendix G. Sample Hosted Pay Page
V1.2.1	October 17, 2008	<ul style="list-style-type: none"> -Section 10. What Information will I get as a Response to My Transaction Request? <ul style="list-style-type: none"> -Added eci Visa/MCSC Definitions.
V1.2.2	September 6, 2011	<ul style="list-style-type: none"> -Section 9. How do I Initiate a Recurring Billing Transaction? <ul style="list-style-type: none"> - Added recurUnit End of Month. -Appendix B. Card Validation Digits (CVD) <ul style="list-style-type: none"> - Added CVD Response for American Express/JCB.
V1.2.3	April 2, 2012	<ul style="list-style-type: none"> -New download link updated in various locations http://www.esselectplus.ca/en/downloadable-content -Section 7-B. The Merchant Resource Center Hosted Paypae Configuration Tool - Configuring the Hosted Pay Page - updated format <ul style="list-style-type: none"> -Added Asynchronous Transaction Response -Added Transaction Risk scoring -Section 10. What information will I get as a response to my transaction Request? <ul style="list-style-type: none"> -Added new variable name (ticket) -Corrected Visa/MCSC definitions for crypt type 6. -Added Section 12. Preload Data -Appendix B. Card Validation Digits(CVD) <ul style="list-style-type: none"> -Added American Express/JCB response codes -Appendix C. Address Verification Service (AVS) <ul style="list-style-type: none"> -Added American Express/JCB response codes -Appendix D. Additional Information for CVD/AVS <ul style="list-style-type: none"> -Added American Express/JCB
V1.2.4	May 28, 2013	<ul style="list-style-type: none"> -New download link updated in various locations https://developer.moneris.com -Section 7-B. The Merchant Resource Center Hosted Pay Page Configuration Tool <ul style="list-style-type: none"> -Added AVS Response -Added VBV Result Code -Added Card Verification -Added Section 8. Hosted Pay Page Process Flow -Added Section 13. Hosted Pay Page Process Flow with Data Preload -Added Section 14. Implementing Enhanced HPP Data Preload -Added Section 15. HPP Process Flow with Asynchronous Transaction Response -Added Section 16. Implementing the Asynchronous Transaction Response -Added Section 17. HPP Process Flow with Transaction Verification

V1.2.5	June 24, 2013	<ul style="list-style-type: none">-Section 7-B. The Merchant Resource Center Hosted Pay Page Configuration Tool<ul style="list-style-type: none">-Added Hosted Pay Page Description-Added Enhanced Cancel button-Added new JCB and Visa Debit logo options-Added Return a Visa Debit card indicator-Added note about receipts sent in html and text format-Added note about ECI value and subsequent follow-on requests-Section 10. What information will I get as a response to my transaction Request?<ul style="list-style-type: none">-Added new return variable name (is_visa_debit)-Section 20. Special Error Codes<ul style="list-style-type: none">-Added cancel code 914-Section 2 -fixed link for developer portal-Section 28 – updated sample Hosted Pay page image to reflect new look- Added Appendix F: CAVV result codes
---------------	---------------	---

Table of Contents

1.	Introduction	6
2.	About This Documentation	6
3.	Skills and System Requirements.....	6
4.	Verified by Visa	6
5.	MasterCard SecureCode	7
6.	What is the Process I will need to follow?	7
7.	The Merchant Resource Centre Hosted Pay Page Configuration Tool.....	7
8.	Hosted Pay Page Process Flow.....	16
9.	Sending a Basic Transaction to the Hosted Pay Page.....	17
10.	How do I Initiate a Recurring Billing transaction?	22
11.	What Information will I get as a Response to My Transaction Request?	24
	What Do I Need to Include in the Receipt?.....	26
12.	Hosted Pay Page Process Flow with Data Preload.....	27
13.	Implementing Enhanced Hosted Pay Page (Data Preload)	28
14.	Hosted Pay Page Process Flow with Asynchronous Transaction Response	31
15.	Implementing the Asynchronous Transaction Response	31
16.	Hosted Pay Page Process Flow with Transaction Verification	33
17.	Sending a Transaction Verification Request	33
18.	Gift Cards.....	35
19.	Special Error Codes	35
20.	How Do I Test My Solution?.....	36
21.	How Do I Activate My Store?	37
22.	How Do I Configure My Store For Production?	37
23.	How Do I Get Help?	37
24.	Appendix A: Example	38
25.	Appendix B. Card Validation Digits (CVD) and Address Verification Service (AVS)	40
26.	Appendix C. XML POST Response for Financial Transaction	41
27.	Appendix D. Sample Receipt	43
28.	Appendix E: Sample Hosted Pay Page.....	44
29.	Appendix F. CAVV Result Code.....	45

****** PLEASE READ CAREFULLY******

You have a responsibility to protect cardholder and merchant related confidential account information. Under no circumstances should ANY confidential information be sent via email while attempting to diagnose integration or production issues. When sending sample files or code for analysis by Moneris staff, all references to valid card numbers, merchant accounts and transaction tokens should be removed and or obscured. Under no circumstances should live cardholder accounts be used in the test environment.

1. Introduction

The Hosted Pay Page was designed as a solution for those merchants that do not wish to handle credit card information, or who do not have an SSL certificate. When a transaction is initiated the transaction specific information is sent to eSELECTplus Hosted Pay Page via an HTTP POST. The cardholder can then securely enter their credit card information. Upon submission eSELECTplus will either generate a receipt on behalf of the merchant, or forward the cardholder and the response back to the merchant's site so that a custom receipt can be created. Upon receiving the response the merchant can perform a Transaction Verification to ensure that the response is from a legitimate transaction.

2. About This Documentation

This document contains a guide for using the Hosted Pay Page configuration tool in the Merchant Resource Centre of eSELECTplus. Also described are the methods for sending credit card transactions and managing the responses using Hosted Pay Page. If you are interested in also being able to accept INTERAC payments via your online application, please refer to the Hosted Pay Page with INTERAC Online Payment document available for download at: <https://developer.moneris.com>

You have the ability to use other integration methods available through eSELECTplus such as an API or Batch File Upload to process follow-on transactions such as Captures, Voids or Refunds which are not available using the Hosted Pay Page. The Merchant Resource Center can also be used to process such follow-on transactions.

3. Skills and System Requirements

In order to use Hosted Pay Page your system will need to have the following:

1. A web server capable of sending and receiving an HTML Post/Get

As well, you will need to have the following knowledge and/or skill-set:

1. Knowledge of creating an HTML webpage and posting forms.
2. If you are selling more than 1 item you will need some knowledge of a client-side scripting language (JavaScript, PHP . . .) to calculate a final charge amount.
3. If you wish to create your own custom receipts, and perform transaction verification you will need knowledge of a server-side scripting language (PHP, Perl, ASP . . .)

Note: If your solution uses an API to perform follow-on transactions, such as Refunds or Captures, you will be required to demonstrate PCI & PA DSS compliance. Please refer to our documentation page for further details about the available API solutions. Also, if developing a payment application intended for a third-party merchant the application will need to conform to the above mentioned compliance standards. Non compliant solutions may prevent the merchant from boarding with Moneris Solutions.

For further information on PCI DSS & PA DSS requirements, please visit <http://www.pcisecuritystandards.org>. To enquire about the card associations' requirements, such as PCI/CISP/PABP, that will pertain to your application please contact our Sales centre as well as review the eSelectplus_PCI_Requirements.pdf document available for download at <https://developer.moneris.com>

4. Verified by Visa

Verified by Visa (VbV) is a program initiated by Visa. Before approving a transaction eSELECTplus and the Bank that issues the Visa credit cards will attempt to authenticate the cardholder through the use of a password, similar to a debit PIN. When an authentication is attempted the merchant is protected from fraud-related chargebacks.

If you have enrolled in Verified by Visa (VbV) with Moneris and eSELECTplus, the Hosted Pay Page will automatically attempt to perform VbV on every Visa transaction.

5. MasterCard SecureCode

MasterCard SecureCode (MCSC) is a new feature offered by MasterCard. Merchants who have enrolled in this program with Moneris and eSELECTplus will be able to offer their customers added protection against unauthorized credit card use, as well as protect themselves from fraud-related chargebacks. Cardholders that have applied for SecureCode with their issuing bank will be able to use this password similar to a debit PIN number for online transactions with participating online merchants.

Before approving a transaction, eSELECTplus and the Bank that issued the MasterCard will authenticate the cardholder through the use of this password. For merchants who have enrolled in SecureCode, the Hosted Pay Page solution will automatically attempt to perform SecureCode verification on every MasterCard transaction.

6. What is the Process I will need to follow?

You will need to follow these steps.

1. Create a Hosted Pay Page Configuration in the test environment of eSELECTplus Merchant Resource Centre
(<https://esqa.moneris.com/mpg/index.php>)
2. Do the required development as outlined in this document
3. Test your solution in the test environment
4. Activate your production store
5. Create and configure your production Hosted Pay Page store in the production Merchant Resource Centre
(<https://www3.moneris.com/mpg>)
6. Make the necessary changes to move your solution from the test environment into production as outlined in this document

7. The Merchant Resource Centre Hosted Pay Page Configuration Tool

Before you can send a transaction to the Hosted Pay Page you will need to configure several settings through the eSELECTplus Merchant Resource Centre (MRC). To log into the MRC test environment go to <https://esqa.moneris.com/mpg> and use one of the following login IDs.

Test IDs		
Store ID	Username	Password
store1	DemoUser	password
store2	DemoUser	password
store3	DemoUser	password
store5	DemoUser	password

A. Creating a Hosted Pay Page Configuration

Once you have successfully logged in, click on the “ADMIN” menu item on the left and then in the submenu that appears click on “HOSTED CONFIG”. In the production environment, an individual must be granted permission to access and alter the configuration. User permissions may be updated by navigating to “ADMIN” and then selecting “MODIFY USER” from the submenu. Once you have found the user to be updated, click on “Set Permissions” and update their Administrative Permissions.

Each eSELECTplus account may have up to five unique Hosted Pay Page configurations. These do not equate to different stores, all transactions will be logged under the same store and will settle into the same bank account. Each configuration can have a different appearance as well as handle responses in varying ways. Because the eSELECTplus test environment is a shared environment there is no limit to the number of configurations assigned to a specific store account. However there is a 30 day time limit where a store configuration will be deleted after 30 days, regardless of use. Please do not alter or delete configurations that were not generated by you.

To create a new Hosted Pay Page configuration click on “Generate a New Configuration”. You will be assigned a Hosted Pay Page ID (ps_store_id) this is the identifier for this unique configuration. You will also be assigned a Hosted Pay Page Token (hpp_key). The Hosted Pay Page ID and Token are sent as part of the transaction request to securely identify your store and the specific configuration.

B. Configuring the Hosted Pay Page

Generate new hpp_key	This allows you to change the Hosted Pay Page Token (hpp_key). Both the “ps_store_id” and “hpp_key” are to be kept secure, though if security were to be compromised, you may generate a new “hpp_key” without having to create a completely new configuration. Please note that after clicking “Generate new HPP key” that your current key will cease to work immediately – there is also no way to retrieve or revert to the old key.
Basic Configuration	
Description	Add a description to easily identify this Hosted Pay Page configuration. This is especially useful when maintaining more than one Hosted Pay Page configuration.
Transaction Type	<p>This defines what type of transaction will be processed.</p> <p><i>Purchase:</i> The cardholder will be charged immediately and funds will be deposited next business day. This is used if your goods and services are shipped/provided within 24 hours.</p> <p><i>Preauthorization:</i> The funds will be locked but will not be settled until a Capture is performed. The Capture will need to be performed via the Merchant Resource Centre or via an API. Preauthorization (PreAuth) is used if the goods and services are not shipped/provided within 24 hours.</p>
Payment Methods	<p>This defines which Payment Methods the Hosted Pay Page will allow, for example, one or all of the methods listed below. The Payment Methods available depend on the store's set-up and these may include Credit Cards, INTERAC Online or Gift Cards.</p> <p><i>Credit Card:</i> This will offer the customer the option of paying with their credit card once they reach the Hosted Pay Page. Please note, when the customer chooses to pay with a credit card, as opposed to INTERAC Online or Gift Cards, the transaction type that will be processed will be the one defined in the default 'Transaction Type' section above.</p> <p><i>Interac Online:</i> This will offer the customer the option of being forwarded to their online banking to approve the transaction. The funds will be debited directly from their bank account. Please note, all INTERAC Online transactions using the Hosted Pay Page will be processed as purchase transactions, there are also unique receipt requirements for INTERAC Online transactions</p> <p><i>Gift Cards:</i> This will offer the customer the option of using up to two gift cards as part of the transaction. All gift card transactions will be processed as purchase transactions. The customer will have the option to check their gift card balance on the Hosted Pay Page.</p>
Response Method	<p>This determines how the transaction response will be handled.</p> <p><u><i>eSELECTplus will generate a receipt:</i></u> Once the transaction is processed eSELECTplus will generate and display a receipt page based on the Pay Page Appearance and Response Data configurations.</p> <p><u><i>Sent to your server as a POST:</i></u> eSELECTplus will use an HTTP POST to send the transaction responses to your web server so that you can customize the receipt or so that other processes may be initiated on your site.</p> <p><u><i>Sent to your server as a POST containing XML:</i></u> eSELECTplus will use an HTTP POST to send the transaction responses back to your web server so that you can customize the receipt or so that other processes may be initiated on your site. The response will be in XML format that will need to be parsed.</p> <p><u><i>Sent to your server as a GET:</i></u> eSELECTplus will redirect the cardholder to a URL on your</p>

	<p>server and attach the response as a URL encoded query string at the end of the URL so that you can customize the receipt or so that other processes may be initiated on your site. Please note that there are limitations imposed by the browser and operating system on the length of a query string.</p> <hr/> <p> NOTE When handling the response (POST or GET), you must be able to dynamically parse the data. In the future, new variables may be added and the order of the response variables may change.</p> <hr/>
Approved URL	If you have chosen to create your own receipt (in <i>Response Method</i>) you will need to specify the URL where the transaction response will be returned when it is approved. All URLs need to be complete – www.example.com is not sufficient – a proper URL is http://www.example.com/response.php – If a URL is missing or improperly typed it may result in a 404 error or a looping page.
Declined URL	If you have chosen to create your own receipt (in <i>Response Method</i>) you will need to specify the URL where the transaction response will be returned when it is declined. This can be the same as the <i>Approved URL</i> . All URLs need to be complete – www.example.com is not sufficient – a proper URL is http://www.example.com/response.php – If a URL is missing or improperly typed it may result in a 404 error or a looping page.
	<hr/> <p> NOTE Click on the “Save Changes” button to save the existing configuration. If the configuration is not saved the current Hosted Pay Page ID (ps_store_id) and Token (hpp_key) will be deleted after approximately 15 minutes.</p> <hr/>
Enhanced Cancel	If you have chosen to use the Enhanced Cancel feature, a full response will be returned to the <i>Declined URL</i> with a response code of '914' and a message of 'cancelled by cardholder'.

PAY PAGE APPEARANCE

Colours and Styles:

The following fields define the colour scheme that is to be used for the Hosted Pay Page as well as the receipts. The colours must be defined in 6 character hex – there is a hex colour chart available by clicking on “Hex Colour Chart” button.

Background Colour	This defines the background colour of the page.
Font Style	This defines what font group will be used for the Hosted Pay Page. We have defined three groups – Arial/Helvetica/SansSerif, Times New Roman/Times/Serif and Courier New/Courier/Mono.
Primary Text Colour	This defines the colour for the majority of the text on the Hosted Pay Page. This must be legible on the chosen background colour.
Company Name Colour	This defines the colour that will be used for your company name.
Header and Footer Highlight Colour:	This defines the colour that will be used for a highlight bar that will appear below the company name and at the bottom of the Pay Page.
Section Divider Colour	The Hosted Pay Page will be divided into several sections depending on what is displayed. A colour bar is used to define the information. This defines the colour of the Primary Bar.
Section Divider Text Colour	Occasionally the primary colour bar will contain text – this defines the colour of the text that will appear in the Primary Bar. Please ensure that the text is legible.
Subsection Divider	The Hosted Pay Page can be divided into several sub-sections depending on what is displayed. A colour bar is used to define the information. This defines the colour of the

Colour	secondary bar that may subdivide information.
Subsection Divider Text Colour	Occasionally the Subsection Divider will contain text – this defines the colour of the text that will appear in the Subsection Divider. Please ensure that the text is legible.
Hosted Pay Page Data Fields	
The following “Display” fields define what is to be displayed on the Hosted Pay Page where the cardholder enters their card information. Some fields are required to be sent by the merchant, others can be set as input boxes on the Hosted Pay Page. We do not check for completion or validity of the information input therefore we suggest that it be passed from the merchant to us.	
Display Item Details	This field indicates whether a listing of items purchased, taxes and shipping costs is to be displayed. In order to display this information it is required that it be sent in the transaction request. Please see request variables to properly send this data.
Display Customer Details	This includes several supplemental data fields that the merchant may pass to the Hosted Pay Page such as a Customer Id, a Customer Email Address, and an additional data field referred to as the Note data.
Display Billing Address	This field indicates whether the client’s billing information is to be displayed. In order to display this information it is required that it be sent in the transaction request. Please see request variables to properly send this data.
Display Shipping Address	This field indicates whether the client’s shipping information is to be displayed. In order to display this information it is required that it be sent in the transaction request. Please see request variables to properly send this data.
Disable Address Input Boxes	This disables the input boxes so that the cardholder cannot alter/add information in the Address, Note and Email fields. When the input boxes are disabled the data will appear as text.
Display Merchant Name	This field indicates whether the Merchant Name should also be displayed on the Hosted Pay Page. The name that will be displayed is the official Merchant Name that Moneris Solutions has associated with the account and the name that the cardholder will see on their credit card statement. It is mandated by industry regulations that the merchant name be displayed on any checkout pages and receipts, but this field may be omitted if the Hosted Pay Page will be loaded within a frame that already displays the merchant name. If you choose to load the Hosted Pay Page within a frame, you are then required to have an SSL certificate.
Buttons	
The following fields define what is to be displayed and the functionality of the buttons located on the Hosted Pay Page and the receipt, if it is to be generated by eSELECTplus.	
Cancel Button Text	This configures the text that is to appear on the cancel button. The cancel button appears on the credit card input page and allows the cardholder to return to your site if they do not wish to complete the transaction.
Cancel Button URL	This configures the URL associated with the cancel button. The cancel button appears on the credit card input page and allows the cardholder to return to your site if they do not wish to complete the transaction. All URLs need to be complete – www.example.com is not sufficient – the proper URL is http://www.example.com – If a URL is missing or improperly typed it may result in a 404 error or a looping page.
Continue Button Text	This configures the text that is to appear on the continue button. The continue button appears on the receipt page that is generated if the option in “Response Methods” is chosen, otherwise this field will be hidden. The continue button allows the client to return to your site after completing a transaction.
Continue	This configures the URL associated with the continue button. The continue button appears

Button URL	on the receipt page that is generated if the option in “Response Methods” is chosen, otherwise this field will be hidden. All URLs need to be complete – www.example.com is not sufficient – the proper URL is http://www.example.com – If a URL is missing or improperly typed it may result in a 404 error or a looping page.
Hide Continue Button	This configures whether the continue button is to appear in the receipt page or not. We do not advise hiding the continue button.
Hosted Pay Page Input Fields	
The following section defines what input fields are to be displayed on the Hosted Pay Page. When an input field is included, it will automatically be mandatory for the customer to fill it in.	
Display CVD input	This defines whether the Hosted Pay Page should include the prompt for the Card Validation Digits (CVD). If this input field is displayed on the Hosted Pay Page it is then mandatory that the cardholder complete this data. This input field only applies to MC, VISA, AmEx and other Credit Card plans that support CVD transactions.
Display AVS input	This defines whether the Hosted Pay Page should include the prompt for the Address Verification Service (AVS) details. If these input fields are displayed on the Hosted Pay Page it is then mandatory that the cardholder complete this data. These input fields only apply to MC, VISA, AmEx and other Credit Card plans that support AVS transactions.
Logos	
This section defines what logos will appear on the payment page. Please note that credit card logos are for display only and will not affect what card types you are able to accept. To begin accepting a new card type, please contact the Service Centre at 1-866-319-7450.	
Credit Card Logos	Allows you to select which logos should appear on the Hosted Pay Page. Check off the appropriate logos; Visa, MasterCard, American Express, Diners, Discover, Sears, JCB. Visa Debit
<hr/> <div style="display: flex; align-items: center;">  <p>Click on the "Save Appearance Settings" button to apply these appearance settings to the Hosted Pay Page. If the appearance configuration is not saved these fields will be returned to their last known saved configuration.</p> </div> <hr/>	
RESPONSE / RECEIPT DATA	
<i>Click on the “Configure Response Fields” button to specify what additional fields you wish to have returned in the transaction response or displayed on the response receipt.</i>	
New to this version is the ability to pass back line items, shipping and billing data that previously had to be passed in as “rvar” in order to be returned to the merchant. We still strongly advise storing customer details on the merchant’s server before passing them to the Hosted Pay Page.	
Response/Receipt Field Configuration	
Return Line Item Details	All line item details will be returned to the response URL in the same manner they were passed to the Hosted Pay Page in the request.
Return Shipping Details	All shipping details will be returned to the response URL in the same manner they were passed to the Hosted Pay Page in the request.
Return Billing Details	All billing details will be returned to the response URL in the same manner they were passed to the Hosted Pay Page in the request.

Return Other Customer Fields	Fields such as cust_id, email, and note will be returned to the response URL in the same manner they were passed to the Hosted Pay Page in the request.
Return ECI value	<p>The ECI value that was used during transaction processing will be returned to the response url. This is used to determine the result of a VBV/MCSC transaction. It is strongly encouraged to review all orders even when it appears a VBV/MCSC authentication was successful or attempted.</p> <hr/> <p> NOTE The ECI (crypt type) value sent in a follow-on transaction request must reflect the ECI value received from the Preauthorization.</p> <p>Example: if the ECI value received from the preauthorization is "5", then the subsequent API capture request must also have it's ECI (crypt type) set to "5".</p>
Return the txn_number	The txn_number for the transaction number is returned in the response. This allows automation of captures, voids and refunds through the use of an API.
Return the VbV Result Code	The VbV result code value from Visa will be returned to the response url. This is used to determine the validity of the VbV transaction data. It is strongly encouraged to review this for all VbV transactions.
Return a Visa Debit card indicator.	A value of true or false is sent back indicating if the card provided by the cardholder was a Visa Debit card.
Return AVS data	The Address Verification data entered by the cardholder on the Hosted Pay Page will be returned to the response url.
Asynchronous Transaction Response	
Perform asynchronous data post	<p>This must be checked for asynchronous data post to be enabled. When the asynchronous data post is enabled the Hosted Pay Page will perform a server to server post of the response data as a secondary method of getting the response data. This does not replace the normal transaction response which will still be sent through the browser as a POST or a GET. This is supplementary and can be used to verify/validate the browser response. If you have enabled the asynchronous data post within your production Hosted Pay Page configuration, you will need to specify the response URL in (HTTPS) format. Self signed certificates will be accepted, but an HTTP address will not work.</p> <hr/> <p>If you send characters that are not supported in any of the variables, the extra transaction details may not be stored, displayed or returned in the response.</p> <p>The order_id allows the following characters: a-z A-Z 0-9 _ - : . @ spaces All other request fields allow the following characters: a-z A-Z 0-9 _ - : . @ \$ = /</p> <p> NOTE If you are using accents they must be sent as html entities (é = &eacute;) If these are being sent as GET please note that they must be URL encoded.</p> <p>Also, please note that if the response is to be sent as a GET the extra transaction details may not always be properly returned. This is due to limitations imposed by the browser and operating system on the length of a query string.</p>
 NOTE	Click on the "Save Response Settings" button to apply these chosen additional fields to the transaction response. If the response configuration is not saved these fields will be returned to their last known saved configuration. Next, click on the "Return to main configuration" button to continue with your Hosted Pay Page configuration setup.

SECURITY FEATURES	
Click on “Configure Security” to add extra security measures to the Hosted Pay Page.	
Referring URL	By adding a URL, you specify that you would like us to check whether the transaction is coming from a location (URL) that you allow. Only POSTs sent from one of the specified URLs will be processed. (It is possible for the Referring URL to be “spoofed” – this is not a guaranteed method of securing your transactions – but it makes it more difficult).
Add URL	<p>Here you can specify up to ten Referring URLs to a max of 255 characters. Each URL needs to be complete and at a registered domain – www.example.com is not sufficient – the proper URL is http://www.example.com/index.html (IP addresses are not supported). After specifying a URL, click on the “Add URL” button to add it to the Allowed URLs list. Once a URL has been added, the “Remove URL” button will become available.</p> <hr/> <p> NOTE To verify your Referring URL, you may POST to https://esqa.moneris.com/HPPDP/myurl.php which will display the URL you are posting from.</p> <hr/>
Transaction Risk Scoring	
Enable Transaction Risk Scoring	This must be checked for transaction risk scoring to be enabled. When Transaction Risk Scoring is enabled the Hosted Pay Page will automatically generate a “session_id” in the transaction request and send this to ThreatMetrix for device profiling. When the response is received, eSelectplus then replies with the transaction information and the transaction risk score. Each transaction can only be verified once. The response to the Transaction Risk Scoring will be sent to your server in the form of a POST or GET only.
Card Verification	
Enable Card Verification	This must be checked for a card verification transaction to be performed. When Card verification is enabled the Hosted Pay page will check the validity of a credit card before it is registered for recurring billing. This will only be performed when 'Bill Now' is set to 'false'.
Transaction Verification	
Enable Transaction Verification	This must be checked for transaction verification to be enabled. When Transaction Verification is enabled the Hosted Pay Page will return a “transactionKey” in the transaction response. When the response is received the fields should be logged and a transaction verification request is sent to eSELECTplus. eSELECTplus then replies with transaction information and whether the transaction was valid or not. Each transaction can only be verified once and it must be verified within 15 minutes of the original transaction being performed. This allows you to ensure that the responses sent to your page are not “spoofed” and that you are only receiving the responses once. If you also intend to check the <i>Referring URL</i> you must ensure that the source of the verification request is in the list of <i>Allowed URLs</i> .
Response Method	<p>This determines how the transaction verification response will be handled.</p> <p><u>Sent to your server as a POST:</u> eSELECTplus will use an HTTP POST to send the transaction verification responses to your web server so that other processes may be initiated on your site.</p> <p><u>Sent to your server as a GET:</u> eSELECTplus will redirect the cardholder to a URL on your server and attach the transaction verification response as a URL encoded query string at the end of the URL so that other processes may be initiated on your site.</p>

	<p><i>Displayed as XML on our server:</i> Once the transaction verification has been performed eSELECTplus will generate a page and display an XML string. This can be used in conjunction with cURL, screen scraping or other such methods.</p> <p><i>Displayed as key/value pairs on our server:</i> Once the transaction verification has been performed eSELECTplus will generate a page and display key value pairs. This can be used in conjunction with cURL, screen scraping or other such methods.</p> <hr/> <p> NOTE When handling the response, you must be able to dynamically parse the data. In the future, new variables may be added and the order of the response variables may change.</p>
<p>Response URL</p>	<p>If you have chosen to have the transaction verification response sent back to you in either a POST or GET (in <i>Response Method</i>) you will need to specify the URL where the transaction response will be returned. The URL needs to be complete and at a registered domain – www.example.com is not sufficient – a proper URL is http://www.example.com/response.php (IP addresses are not supported). If a URL is missing or improperly typed it may result in a 404 error or a looping page. If you have chosen to have eSELECTplus display an XML string or key/value pairs this field may be left blank.</p>
<p> NOTE</p>	<p>Click on the “Save Verification Settings” button to apply these chosen additional security features to the Hosted Pay Page. If the security feature is not saved these fields will be returned to their last known saved configuration. Next, click on the “Return to main configuration” button to continue with your Hosted Pay Page configuration setup.</p>
<p>EMAIL RECEIPTS</p>	
<p>Click on “Configure Email Receipts” to specify email receipt conditions and appearance.</p> <hr/> <p>New to this version. All emails are now sent in HTML and text format. This change will allow recipient’s email client to display their receipt in their default preferred format. This will allow better formatting for customer receipts which will accommodate both web and mobile receipts.</p> <p> NOTE Although we are sending in both HTML and text format, the card holder will only receive one receipt.</p> <p>All emails now come from the moneris.com mail server vs. the www3.moneris.com mail server which had been identified as being a problem for certain spam filters.</p>	
<p>Receipt Conditions</p>	
<p>Send email to cardholder if transaction approves</p>	<p>This defines whether a receipt is to be e-mailed to the cardholder if the transaction approves. If this option is selected, but the cardholder’s e-mail address is not sent in the POST, then the receipt will not be emailed out. Please refer to Section 8 – Other Transaction Details for an example of how to send the cardholder’s email address (“email”) in the request.</p>
<p>Send email to cardholder if transaction declines</p>	<p>This defines whether a receipt is to be e-mailed to the cardholder if the transaction declines. If this option is selected, but the cardholder’s e-mail address is not sent in the POST, then the receipt will not be emailed out. Please refer to Section 8 – Other Transaction Details for an example of how to send the cardholder’s email address (“email”) in the request.</p>
<p>Send email to merchant if transaction approves</p>	<p>This defines whether a receipt is to be e-mailed to the merchant if the transaction approves. If this option is selected, but the merchant’s e-mail address is not provided in the “Merchant email address” field, then the receipt will not be emailed out.</p>
<p>Send email to</p>	<p>This defines whether a receipt is to be e-mailed to the merchant if the transaction declines. If</p>

merchant if transaction declines	this option is selected, but the merchant's e-mail address is not provided in the "Merchant email address" field, then the receipt will not be emailed out.
Include 'rvar' in merchant email	"rvar"s can be a series of variables/values that will be echoed back in the transaction response. This field indicates whether these fields are to be included in the email to the merchant. They will NOT be included in the email to the client. If this option is selected, but the merchant's e-mail address is not provided in the "Merchant email address" field, then the "rvar"s will not be e-mailed to the merchant.
Merchant email address	This field allows you to provide the e-mail address you want all the Merchant Email Receipts to be sent to, as defined in the 3 options mentioned above. Only one e-mail address may be provided.

Receipt Appearance

Include Line Item Details	This field indicates whether a listing of items purchased, taxes and shipping costs is to be displayed. In order to display this information it is required that it be sent in the transaction request. Please see request variables to properly send this data.
Include Billing Details	This field indicates whether the client's billing information is to be displayed. In order to display this information it is required that it be sent in the transaction request. Please see request variables to properly send this data.
Include Shipping Details	This field indicates whether the client's shipping information is to be displayed. In order to display this information it is required that it be sent in the transaction request. Please see request variables to properly send this data.
Include Customer Details	This will include the cust_id, client email address, and the note field data.
Email Text	New to version 3 of the Hosted Pay Page is the ability to add a short message that will appear at the top of the email receipts. There is a 255 character limit and the characters supported are limited to letters, numbers and the following characters ' # @ _ , - . and space and enter (newline).



NOTE

If you send characters that are not supported in any of the variables, the extra transaction details may not be stored or included in the email receipt.

The order_id allows the following characters: **a-z A-Z 0-9 _ - : . @ spaces**

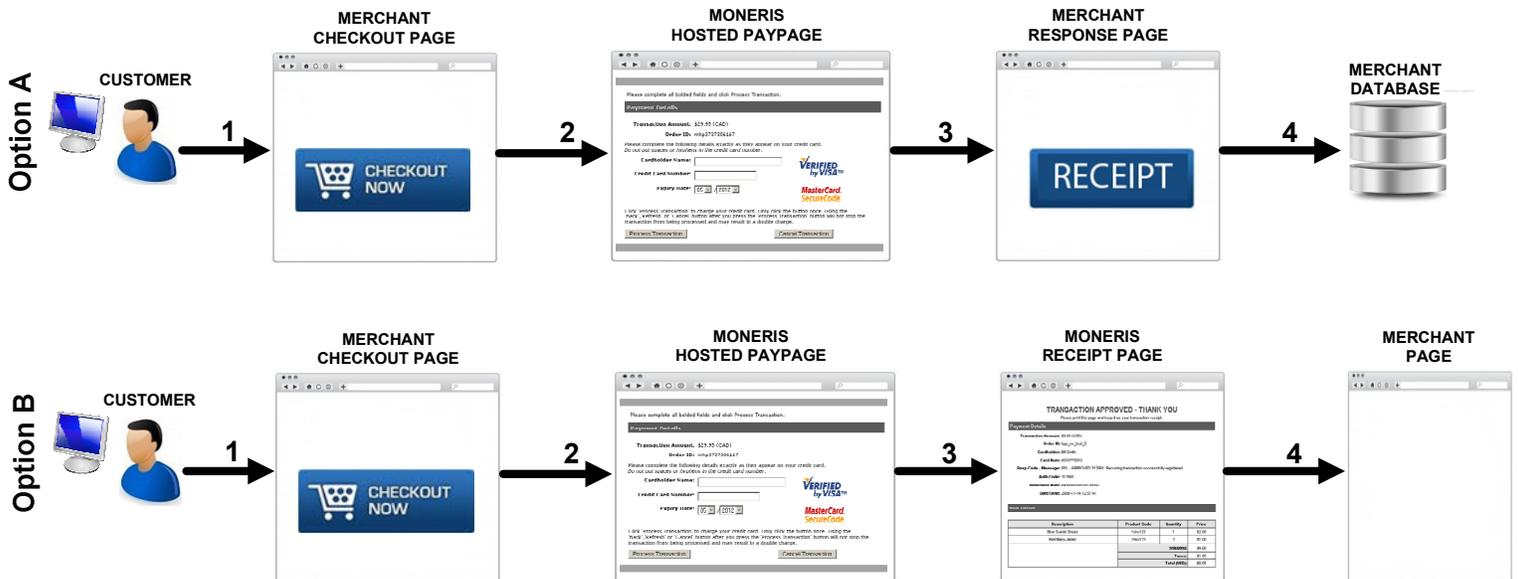
All other request fields allow the following characters: **a-z A-Z 0-9 _ - : . @ \$ = /**



NOTE

Click on the "Save Email Settings" button to apply these chosen Email Receipt settings to the Hosted Pay Page. If the Email Receipt configuration is not saved these fields will be returned to their last known saved configuration. Next, click on the "Return to main configuration" button to continue with your Hosted Pay Page configuration setup.

8. Hosted Pay Page Process Flow



Outlined above is the basic Hosted Pay Page flow which allows a cardholder to process a transaction. There are 2 options available:

- A. This option is for merchants that choose to have the response sent back to them in either a POST or GET format so that they may build the receipt themselves, as well as store the response variables as needed.
- B. This option is for merchants that choose to have Moneris generate the receipt.

The steps required to achieve this are as follows:

1. The customer arrives on the merchant's website (application). At this point the merchant must determine the amount of the transaction and also collect any additional data.
2. Once the cardholder is ready to pay, the merchant's checkout page will submit an HTTP form POST to the Moneris Hosted Pay Page (HPP). At this time, the customer will be redirected from the merchant's website to the Moneris Hosted Pay Page URL. For details on the HTTP form POST, please refer to section 9 of this document which outlines the mandatory and optional fields that may be sent to the Hosted Pay Page.

Option A

3. On the Hosted Pay Page, the cardholder will fill in their secure payment details such as their card number and submit the transaction. At this time, Moneris will process the transaction and then build a response to send back to the merchant in a POST or GET format to the Response URL provided in the Hosted Pay Page Configuration as described in 7B above.
4. Once the merchant receives the response details they must provide a receipt to the customer and then may store these details for future purposes such as reporting or tracking.

Option B

3. On the Hosted Pay Page, the cardholder will fill in their secure payment details such as their card number and submit the transaction. At this time, Moneris will process the transaction and then display a receipt to the customer.
4. Once the customer is ready to continue they will then be redirected back to the merchant's website to Response URL provided in the Hosted Pay Page Configuration as described in 7B above.

9. Sending a Basic Transaction to the Hosted Pay Page

The customization options offered by the Hosted Pay Page make it difficult to give a clear cut example of a transaction. Below are a series of tables containing all the fields that can be sent in a Hosted Pay Page request. The first table contains the required variables – these must be sent. Subsequent tables contain variables that can be sent optionally. The appearance and functionality of the Hosted Pay Page is controlled by the Hosted Pay Page Configuration tool located in the Merchant Resource Centre outlined above.

A. Required Variables

Required Variables		
Variable Name	Type	Description
	form	https://esqa.moneris.com/HPPDP/index.php - Development https://www3.moneris.com/HPPDP/index.php - Production
ps_store_id	hidden	Provided by Moneris Solutions – Hosted Pay Page Configuration Tool. Identifies the configuration for the Hosted Pay Page.
hpp_key	hidden	Provided by Moneris Solutions – Hosted Pay Page Configuration Tool. This is a security key that corresponds to the ps_store_id.
charge_total	hidden	Final purchase Amount - no \$, must include 2 decimal places (i.e. 3.00)

Sample

Below is a sample of the most basic Hosted Pay Page call using only the required variables. This code will create a submit button that will send a charge of \$1.00 to a store with a configuration ID of AF4Fs1024.

```
<FORM METHOD="POST" ACTION=https://esqa.moneris.com/HPPDP/index.php>
  <INPUT TYPE="HIDDEN" NAME="ps_store_id" VALUE="AF4Fs1024">
  <INPUT TYPE="HIDDEN" NAME="hpp_key" VALUE="Hsjh4GSr4g">
  <INPUT TYPE="HIDDEN" NAME="charge_total" VALUE="1.00">
  <!--MORE OPTIONAL VARIABLES CAN BE DEFINED HERE -->
  <INPUT TYPE="SUBMIT" NAME="SUBMIT" VALUE="Click to proceed to Secure Page">
</FORM>
```

B. Item Details – Optional

Item Details can be sent so that they may be presented in the receipt or email that eSELECTplus generates. eSELECTplus will also store the item data so that it may be reviewed through the eSELECTplus Merchant Resource Centre. eSELECTplus does not perform any calculations to arrive at a final amount.

Optional Variable - Item Details

Where "n" is an alphanumeric value less than 10 characters long, unique to each item. For each item all five variables should be included. These items will be stored in the Merchant Resource Centre and will be included in the email receipt if "Include Line Item Details" is selected in the Email Receipt Configuration and will be displayed on the payment page if "Display Items Details" is selected in the Appearance Configuration.

Variable Name	Type	Description
id n	hidden	Product Code - SKU (max 10 chars)
description n	hidden	Product Description - (max 15 chars)
quantity n	hidden	Quantity of Goods Purchased - (max - 4 digits)
pric n	hidden	Unit Price - (max - "7"."2" digits, i.e. min 0.00 & max 9999999.99)
subtotal n	hidden	Quantity X Price of Product - (max - "7"."2" digits, i.e. min 0.00 & max 9999999.99)

NOTE: you must send a quantity**n** > 0 or the item will not be added to the item list.

Sample

The code below will create 1 item and define its details. It will set the product id**n**, description**n**, quantity**n**, pric**n** and subtotal**n** details for the item to be displayed in the receipt.

```
<input type="hidden" name="id1" value="1">
<input type="hidden" name="description1" value="Blue Suede Shoes">
<input type="hidden" name="quantity1" value="3">
<input type="hidden" name="pric1" value="40.00">
<input type="hidden" name="subtotal1" value="120.00">
```

C. Other Transaction Details

These miscellaneous fields can be submitted as part of the transaction request.

Optional Variables – Transaction Details		
Variable Name	Type	Description
cust_id	hidden	This is an ID field that can be used to identify the client, commonly used for student #s, policy #s, client name or invoice #s. Can not be more than 50 chars.
order_id	hidden	MUST be unique per transaction and be no more than 50 chars. System will generate if excluded.
lang	hidden	This defines what language the Hosted Pay Page and the receipts will be in: en-ca = English fr-ca = French If the tag is not included the hosted Pay Page will default to English.
gst	hidden	This is where you would include Goods and Services Tax charged, should you wish it to be displayed on the items list. (min 0.00 & max 9999999.99)
pst	hidden	This is where you would include Provincial Sales Tax charged, should you wish it to be displayed on the items list. (min 0.00 & max 9999999.99)
hst	hidden	This is where you would include Harmonized Sales Tax charged, should you wish it to be displayed on the items list. (min 0.00 & max 9999999.99)
shipping_cost	hidden	This is where you would include shipping charges, should you wish it to be displayed on the items list. (min 0.00 & max 9999999.99)
note	text	This is any special instructions that you or the cardholder might like to store. Can not be more than 50 chars.
email	text	Customer's email address. This address will be used for the email receipts. Can not be more than 50 chars. If you have chosen to send an email receipt to the cardholder this field must be included.
eci	hidden	If using the Hosted Pay Page to integrate an internal order management system for Mail/Telephone Orders, send an eci value of 1. If the Hosted Pay Page supports VbV/MCSC, the eci generated by VbV/MCSC will override the value passed in.



NOTE

The order_id allows the following characters: **a-z A-Z 0-9 _ - : . @ spaces**

All other request fields allow the following characters: **a-z A-Z 0-9 _ - : . @ \$ = /**

Sample

The code below will set the cust_id, order_id, and the lang as well as add gst, pst, hst and shipping cost details for display in the receipt.

```
<INPUT TYPE="HIDDEN" NAME="cust_id" VALUE="invoice: 123456-12-1">
<INPUT TYPE="HIDDEN" NAME="order_id" VALUE="oid43333">
<INPUT TYPE="HIDDEN" NAME="lang" VALUE="fr-ca">
<INPUT TYPE="HIDDEN" NAME="gst" VALUE="0.80">
<INPUT TYPE="HIDDEN" NAME="pst" VALUE="0.70">
<INPUT TYPE="HIDDEN" NAME="hst" VALUE="1.50">
<INPUT TYPE="HIDDEN" NAME="shipping_cost" VALUE="4.99">
<INPUT TYPE="HIDDEN" NAME="eci" VALUE="1">
```

D. Shipping and Billing Information

Shipping and billing information will be stored in the Merchant Resource Centre. It may also be included in the email receipt if "Include billing details" and/or "Include shipping details" are selected in the Email Receipt Configuration. Also, these fields may be returned in the transaction response or displayed on the response receipt if "Return billing details." and/or "Return shipping details." are selected in the Response/Receipt Field Configuration.

Shipping and Billing Address Variables

Note: Each of the fields below is alphanumeric and can not be more than 30 characters.

Variable Name	Type	Variable Name	Type
ship_first_name	text	bill_first_name	text
ship_last_name	text	bill_last_name	text
ship_company_name	text	bill_company_name	text
ship_address_one	text	bill_address_one	text
ship_city	text	bill_city	text
ship_state_or_province	text	bill_state_or_province	text
ship_postal_code	text	bill_postal_code	text
ship_country	text	bill_country	text
ship_phone	text	bill_phone	text
ship_fax	text	bill_fax	text

Sample

The code below will set the billing and shipping details so they may be stored in the Merchant Resource Centre. Also, depending on the Hosted Pay Page configuration, these fields may also be included in the response and receipt.

```
<INPUT TYPE="HIDDEN" NAME="bill_first_name" VALUE="John">
<INPUT TYPE="HIDDEN" NAME="bill_last_name" VALUE="Smith">
<INPUT TYPE="HIDDEN" NAME="bill_company_name" VALUE="Moneris Solutions">
<INPUT TYPE="HIDDEN" NAME="bill_address_one" VALUE="101 Main St">
<INPUT TYPE="HIDDEN" NAME="bill_city" VALUE="Smallville">
<INPUT TYPE="HIDDEN" NAME="bill_state_or_province" VALUE="NT">
<INPUT TYPE="HIDDEN" NAME="bill_postal_code" VALUE="Z1Z 1Z1">
<INPUT TYPE="HIDDEN" NAME="bill_country" VALUE="Canada">
<INPUT TYPE="HIDDEN" NAME="bill_phone" VALUE="555-555-5555">
<INPUT TYPE="HIDDEN" NAME="bill_fax" VALUE="555-555-6666">

<INPUT TYPE="HIDDEN" NAME="ship_first_name" VALUE="Jen">
<INPUT TYPE="HIDDEN" NAME="ship_last_name" VALUE="Smith">
<INPUT TYPE="HIDDEN" NAME="ship_company_name" VALUE="Moneris Solutions">
<INPUT TYPE="HIDDEN" NAME="ship_address_one" VALUE="150 Lakeshore Rd">
<INPUT TYPE="HIDDEN" NAME="ship_city" VALUE="Springfield">
<INPUT TYPE="HIDDEN" NAME="ship_state_or_province" VALUE="IL">
<INPUT TYPE="HIDDEN" NAME="ship_postal_code" VALUE="234567">
<INPUT TYPE="HIDDEN" NAME="ship_country" VALUE="USA">
<INPUT TYPE="HIDDEN" NAME="ship_phone" VALUE="333-555-5555">
<INPUT TYPE="HIDDEN" NAME="ship_fax" VALUE="333-555-6666">
```



NOTE

If you send characters that are not included in the allowed list, the extra transaction details (ex. bill_ or ship_) may not be stored, included in the receipt or in the response. Please refer to the section 8-C for the list of allowed characters.

Also, the data sent in Billing and Shipping Address variables will not be used for any address verification. Please refer to section **Error! Reference source not found.** for information on Address Verification Service

(AVS).

E. Optional 'rvar' Variables

Optional Variables – "rvar" Variables

Where "n" is an alphanumeric value less than 10 characters long, unique to each rvar variable. The data sent in the rvar variables will NOT be stored in the Merchant Resource Center. These fields will be echoed back in the transaction response – in a GET or POST method. Also, they may be sent in the email receipt to the merchant if "Include 'rvar' in merchant email" is selected in the Email Receipt Configuration.

Variable Name	Type	Description
<i>rvarn</i>	hidden	If these extra variables are sent in the request, they will be echoed back in the response (if GET or POST have been selected for the <i>Response Method</i>). Commonly used for session ID's. These variables must begin with "rvar" and then contain any alphanumeric string (i.e. rvar1, rvarname, rvarMyVariable).

Sample

The code below will send 3 rvar's in the request so that they may be returned in the response or displayed on the merchant's email receipt.

```
<INPUT TYPE="HIDDEN" NAME="rvar1" VALUE="1">
<INPUT TYPE="HIDDEN" NAME="rvar_monkey" VALUE="monkeys are funny">
<INPUT TYPE="HIDDEN" NAME="rvar_123abc" VALUE="123abc">
```

10. How do I Initiate a Recurring Billing transaction?

eSELECTplus offers an optional feature to process your regular recurring payments. This is often used for subscriptions, memberships or any time a fixed amount is charged at a regular interval. The transaction is sent to eSELECTplus using the same methods listed above and the fields listed below are added to the POST. Recurring Billing must be enabled on your merchant account – please contact Moneris sales to have this feature added to your profile if you have not already done so.

Recurring Variables (all required for a recurring billing transaction)		
Variable Name	Type	Description
doRecur	hidden	1 = initiate a Recurring transaction, anything else will not initiate a recurring transaction
recurUnit	hidden	Must be "day", "week" "month" or "eom". This is the base unit for the recurring interval.
recurPeriod	hidden	Numeric value. The period is used in conjunction with recurUnit to determine the interval between payments. Example: recurUnit= "month" and recurPeriod = "12" the charge will be billed once every twelve months. If recurUnit= "week" and recurPeriod = "1" the charge will be billed once a week. If recurUnit = "eom" and recurPeriod = "3" the charge will be billed every 3 months (on the last day of the month). Please note that the total duration of the recurring billing transaction should not exceed 5-10 years in the future.
recurStartDate	hidden	Must be in the format "YYYY/MM/DD" – this determines the date of the first charge. This date must be in the future – it cannot be the date the transaction is sent (Please see recurStartNow to bill card holder immediately).
recurStartNow	hidden	"true" / "false" This will charge the transaction immediately and then initiate recurring billing to commence on the recurStartDate. Example: to charge a card immediately (assuming the date is May 9 2006) and then bill every month thereafter: recurStartNow = "true" recurStartDate = "2006/05/09" - set the date 1 month after the present recurUnit = "month" recurPeriod = "1"
recurAmount	hidden	Amount to charge on a recurring basis - no \$, must include 2 decimal places (i.e. 3.04). This can vary from the charge_total. If using recurStartNow charge_total is used for the immediate transaction and recurAmount is used for every transaction in the future. Example: A member is joining halfway through May 2006 – you would like to bill the remaining half of the month (\$20.00) and then bill them on the first day of the month every month for the full month (\$40.00) starting June 1st 2006. charge_total="20.00" recurAmount="40.00" recurStartNow="true" recurStartDate="2006/06/01" recurUnit="month" recurPeriod="1"
recurNum	hidden	The number of times to process the recurring charge. Example: if you are billing a client monthly for one year recurNum="12" We advise against setting a period of longer than 5 years. The suggested maximum should be calculated using your recurUnit and recurPeriod settings over a 5 year period. Example: If you would like to bill a client indefinitely you should set the recurNum to be approximately 5 years in the future – so if you are billing once a month

recurNum="60" (12 x 5) or if you are billing every two weeks recurNum = "130"
(26 x 5)

**NOTE**

When completing the recurring billing portion please keep in mind that to prevent the shifting of recur bill dates, avoid setting the start_date for anything past the 28th of any given month when using the recur_unit set to "month". For example, all billing dates set for the 31st of May will shift and bill on the 30th in June and will then bill the cardholder on the 30th for every subsequent month. To set the billing dates for the end of the month please set the recur_unit to "eom"..

Sample

Below is a sample of the HTML to initiate a recurring transaction – this must be sent as part of a basic Hosted Pay Page request that includes: hpp_key, ps_store_id and charge_total. This transaction will bill the amount sent in charge_total immediately (using the example above charge_total="1.00") Then commencing June 1st 2006 (assuming a date in the future) the card will be billed \$4.00 every 2 weeks, 26 times (one year)

```
<input type="hidden" name=doRecur value ="1">  
<input type="hidden" name=recurUnit value ="week">  
<input type="hidden" name=recurStartDate value="2006/06/01">  
<input type="hidden" name=recurNum value="26">  
<input type="hidden" name=recurStartNow value ='true'>  
<input type="hidden" name=recurPeriod value='2'>  
<input type="hidden" name=recurAmount value='4.00'>
```

11. What Information will I get as a Response to My Transaction Request?

For each transaction you will receive a response message. The fields that will be included in the response are indicated in the table below.

The Receipt can be handled in two ways depending on how the “Response Method” has been configured.

1. eSELECTplus can generate a receipt on your behalf and present it to the client. The receipt will be relatively generic in appearance and will be based on the settings from the Hosted Pay Page Configuration in the Merchant Resource Centre. Please refer to section 7-B – Response/Receipt Data to configure the receipt. If you are using this method there is no need to review this section or section 0.

2. The receipt values will be sent back to the URL specified in the Hosted Pay Page Configuration settings from the Merchant Resource Centre. You can then create a custom receipt or use it to initiate a secondary process. These values can be passed back appended to the URL in a query string format or as an HTTP POST.

Response Fields		
Variable Name	Size/Type	Description
response_order_id	50 / an	order_id specified in request or generated by Hosted Pay Page
response_code	3 / an	Transaction Response Code < 50: Transaction approved >= 50: Transaction declined NULL: Transaction was not sent for authorization
		If you would like further details on the response codes that are returned please see the Response Codes document available for download at: https://developer.moneris.com
date_stamp	yyyy-mm-dd	Processing host date stamp
time_stamp	##:##:##	Processing host time stamp
bank_approval_code	8 / an	Authorization code returned from the issuing institution
result	1 / num	1 = approved , 0 = declined, incomplete
trans_name	an	Type of transaction that was performed purchase: cardholder was billed immediately preauth: funds were locked on the card – a capture will need to be performed to have the funds deposited into merchant’s account (see Merchant Resource Centre User’s Guide). A PreAuth transaction must be reversed if it is not to be captured. To reverse the full amount of the PreAuth, please use the Capture transaction with a dollar amount of “0.00”. cavv_purchase: similar to purchase but a VBV/MCSC authentication attempt was made. cavv_preauth: similar to preauth but a VBV/MCSC authentication attempt was made.
cardholder	40 / an	Cardholder’s name
charge_total	n.nn (2dec)	Amount of the transaction
Card	2 / alpha	Credit Card Type M = Mastercard V = Visa AX = American Express DC = Diners Card NO = Novus / Discover SE = Sears
f4l4	####*#####	First 4 and last 4 digits of the card #

message	100 / an	Response description returned from issuing institution or from eSELECTplus if there is a system error.								
iso_code	2 / num	ISO response code								
bank_transaction_id	18 / num	The reference number is an 18 character string that references the terminal used to process the transaction as well as the shift, batch and sequence number. This data is typically used to reference transactions on the host systems and must be displayed on any receipt presented to the customer. This information should be stored by the merchant. The following illustrates the breakdown of this field where "660123450010690030" is the reference number returned in the message, "66012345" is the terminal id, "001" is the shift number, "069" is the batch number and "003" is the transaction number within the batch. Moneris Host Transaction identifier								
transactionKey	100/an (optional)	This is an encrypted string that is returned when using the transaction verification feature. There is no need to decrypt the string. It needs to be passed back to eSELECTplus to verify the authenticity of the transaction.								
Ticket	an	The value returned from the preload data request.								
Rvarn	optional	These extra variables can be sent in the request and will be echoed back in the response. These variables must begin with "rvar" and then contain any alphanumeric string (i.e. rvar1, rvarname, rvarMyVariable). If they are not posted in the request, they will not be included in the response.								
Eci	1/num	Electronic Commerce Indicator that was sent with the transaction. Possible values are:								
<table border="1"> <thead> <tr> <th>Crypt Type</th> <th>Visa/MCSC Definitions</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>- Fully authenticated - There is a liability shift and the merchant is protected from chargebacks.</td> </tr> <tr> <td>6</td> <td>- VB/MCSC has been attempted - VB -There is a liability shift and the merchant is protected from chargebacks -MCSV –No liability shift and the merchant is not protected from chargebacks.</td> </tr> <tr> <td>7</td> <td>- Non-VB/MCSC transaction - Merchant is no longer protected from chargebacks</td> </tr> </tbody> </table>			Crypt Type	Visa/MCSC Definitions	5	- Fully authenticated - There is a liability shift and the merchant is protected from chargebacks.	6	- VB/MCSC has been attempted - VB -There is a liability shift and the merchant is protected from chargebacks -MCSV –No liability shift and the merchant is not protected from chargebacks.	7	- Non-VB/MCSC transaction - Merchant is no longer protected from chargebacks
Crypt Type	Visa/MCSC Definitions									
5	- Fully authenticated - There is a liability shift and the merchant is protected from chargebacks.									
6	- VB/MCSC has been attempted - VB -There is a liability shift and the merchant is protected from chargebacks -MCSV –No liability shift and the merchant is not protected from chargebacks.									
7	- Non-VB/MCSC transaction - Merchant is no longer protected from chargebacks									
txn_num	20/an	Gateway Transaction identifier. This value is required if merchant decides to send automated captures, voids or refunds through an API.								
avs_response_code	1/an	Indicates the address verification result. Refer to Error! Reference source not found. for further details. To test AVS you must create a configuration in "store5" and use that configuration for testing.								
cvd_response_code	1/an	Indicates the CVD validation result. Refer to for further details. To test CVD you must create a configuration in "store5" and use that configuration for testing.								
cavv_result_code	1 / an	The Cardholder Authentication Verification Value (CAVV) is a value that allows VisaNet to validate the integrity of the VbV transaction data. These values are passed back from the issuer to the merchant after the VbV/SecureCode authentication has taken place. For example: If the ECI returned is a "6" and the result code is a "B", it becomes liable for chargeback. Please see appendix F for the CAVV result codes table								
is_visa_debit	boolean	A value of 'true' or 'false' is sent back which indicates if the card provided by the cardholder was a Visa Debit card.								

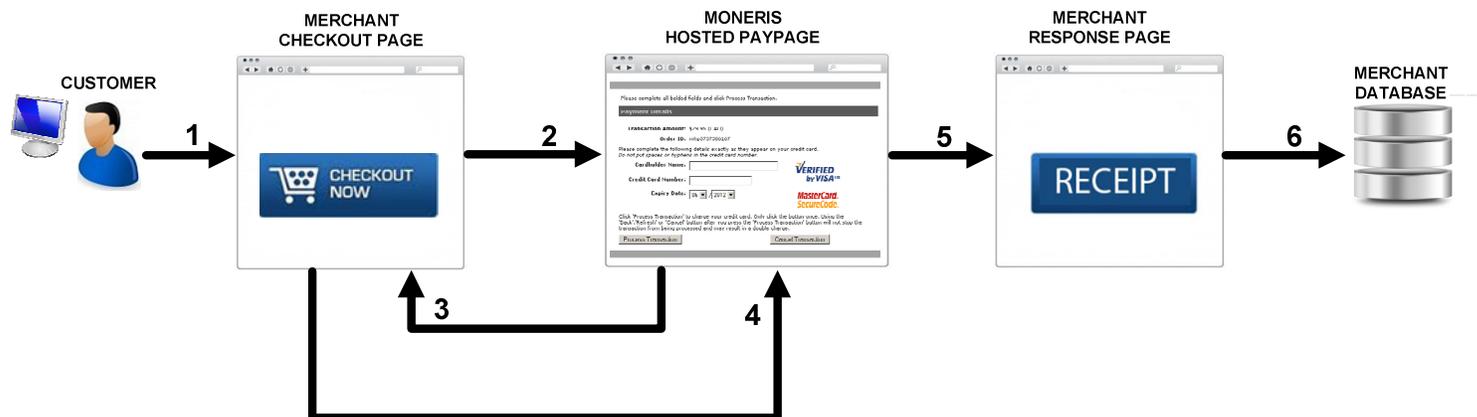
 **NOTE** To determine if a transaction is approved the response_code will have a value of less than 50. If it is declined the response_code will be 50 or greater. A value of NULL means the transaction was incomplete.

What Do I Need to Include in the Receipt?

Visa and MasterCard expect certain variables be returned to the cardholder and presented as a receipt when a transaction is approved. If the Hosted Pay Page is configured to return the response to your webserver it is imperative that you display the information listed below. These required fields are listed below and includes the corresponding variable name as returned by the Hosted Pay Page or a proper description in brackets.

1. **Amount** - (charge_total)
2. **Transaction Type** - (trans_name)
3. **Date and Time** - (date_stamp & time_stamp)
4. **Authorisation Code** - (bank_approval_code)
5. **ResponseCode** - (response_code)
6. **ISO Code** - (iso_code)
7. **Response Message** - (message)
8. **Reference Number** - (bank_transaction_id)
9. **Goods and Services Order** - (description of the products / services ordered)
10. **Merchant Name** - (Your Business Name – should be same as what you registered with Moneris Solutions)
11. **Merchant URL** - (Your business website)
12. **Cardholder Name** - (cardholder)
13. **Return Policy** (only a requirement for e-commerce transactions)

12. Hosted Pay Page Process Flow with Data Preload



Outlined above is the Hosted Pay Page flow with the Data Preload feature implemented.

The steps required to achieve this are as follows:

1. The customer arrives on the merchant's website (application). At this point the merchant must determine the amount of the transaction and also collect any additional data.
2. Once the cardholder is ready to pay, the merchant's checkout page will submit an HTTPS POST using a server side programming language to the Moneris Hosted Pay Page (HPP) sending over all of the transaction data. NOTE: the cardholder will not be redirected yet.
3. The Hosted Pay Page (HPP) will store this transaction data and respond back to the merchant's site by sending an XML response containing a ticket number.
4. The merchant's checkout site will need to collect this response data, and build a new form POST to the Moneris Hosted Pay Page. At this time, the customer will be redirected from the merchant's website to the Moneris Hosted Pay Page URL.
5. On the Hosted Pay Page, the cardholder will fill in their secure payment details such as their card number and submit the transaction. At this time, Moneris will process the transaction and build a response.
6. Once the merchant receives the response details they must provide a receipt to the customer and then may store these details for future purposes such as reporting or tracking.

For details on the Preload form POST data, please refer to section 13 of this document which outlines the mandatory fields.

13. Implementing Enhanced Hosted Pay Page (Data Preload)

This feature allows merchants to preload transaction data into the Hosted Pay Page through a direct server to server request. The Hosted Pay Page then returns an XML response containing a "ticket". The ticket is then sent with the ps_store_id through the shopper's browser in a request to the Hosted Pay Page. The Hosted Pay Page will then link the preloaded data to the browser request by using the ticket information after which the process will proceed as a regular Hosted Pay Page transaction. In a typical shopping cart checkout experience you must manage and ensure that preload requests correspond to the correct shopper's browser session.

Overview of the Preload Process

1. Preload Data Request
2. Handling the Preload Response
3. Proceed to Hosted Pay Page
4. Normal Hosted Pay Page Process

Step 1 – Preload Data Request

The fields below need to be sent via an HTTPS POST using a server side programming language such as .NET, Java, and PHP. Other optional variables could also be passed in this step, such as order ID, customer ID, billing and shipping information, etc.

Data Preload Request Fields	
Variable Name	Description
URL's to POST to	https://esqa.moneris.com/HPPDP/index.php - Development https://www3.moneris.com/HPPDP/index.php - Production
ps_store_id	Provided by Moneris Solutions – Hosted Pay Page Configuration Tool
hpp_key	Provided by Moneris Solutions – Hosted Pay Page Configuration Tool
hpp_preload	Blank value. This is used to indicate that the transaction is a preload transaction
charge_total	Amount to charge, must be have 2 decimal places
order_id	(Optional) Merchant defined unique transaction identifier - must be unique for every transaction attempt. Characters allowed for Order ID: a-z A-Z 0-9 _ - : . @ spaces <i>Note: only the last 10 characters of the Order ID will appear in the Moneris Merchant Direct report</i>

Step 2 - Handling the Preload Response:

After you send the preload data request to Moneris you will receive an XML response with: the hpp_id, the ticket, the order_no, and the response code. The combination of the hpp_id, ticket, and the order ID will uniquely identify this particular set of preloaded data with what's stored already on the Moneris side. To make this work you must ensure the shoppers browser session is linked to the correct ticket

Data Preload Response Fields

Variable Name	Description
hpp_id	The ps_store_id is returned as the hpp_id in the XML response.
ticket	The unique value assigned to the preload transaction
order_id	The order_id that was sent in the preload request; if no order id is sent in the preload request then a unique Order ID will be assigned by the Moneris system.
response_code	Transaction Response Code < 50: data successfully loaded >= 50: data not loaded

Example XML Response

Below is a sample of the data preload Response displayed on our server in XML format

Valid Response:

```
<?xml version="1.0" standalone="yes"?>
<response>
  <hpp_id>4YAHJqa002</hpp_id>
  <ticket>hppEzskxQTZe8DbO80ga</ticket>
  <order_id>hpp_test_1</order_id>
  <response_code>1</response_code>
</response>
```

Step 3 - Proceed to Hosted Pay Page

Upon receiving confirmation from the user that they are ready to proceed, you would then redirect the shopper via an HTTPS post with the variables below to the Moneris Hosted Pay Page.

Proceed to Hosted Pay Page Fields

Variable Name	Description
URL's to POST to	https://esqa.moneris.com/HPPDP/index.php - Development https://www3.moneris.com/HPPDP/index.php - Production
hpp_id	Provided by Moneris Solutions – Hosted Pay Page Configuration Tool
hpp_preload	Blank value. This is used to indicate that the transaction is a preload transaction
ticket	A value returned by the preload request which helps identify the transaction

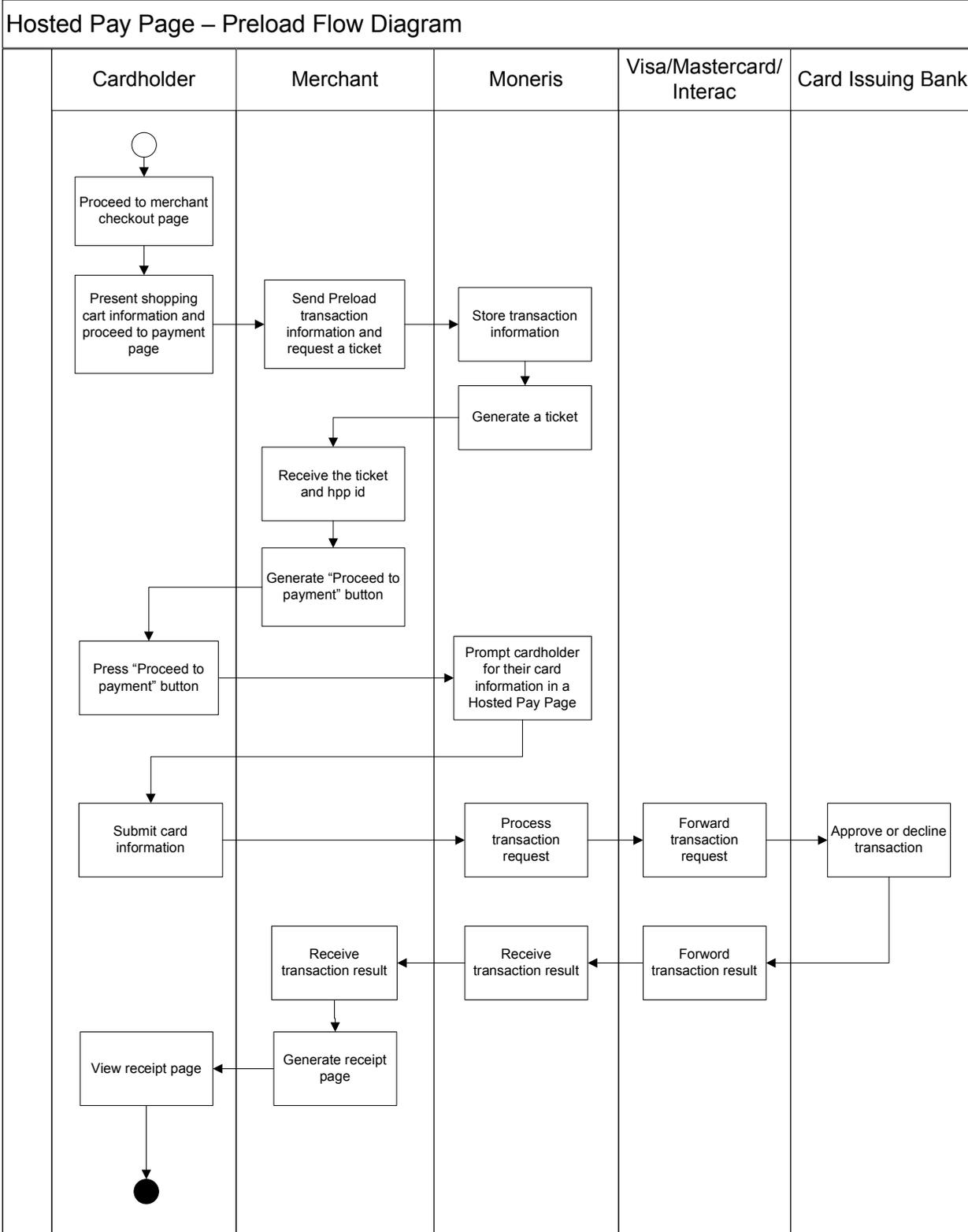
Note: Other optional variables should not be sent, only the above variables should be sent in this request.

Example payment button HTML code:

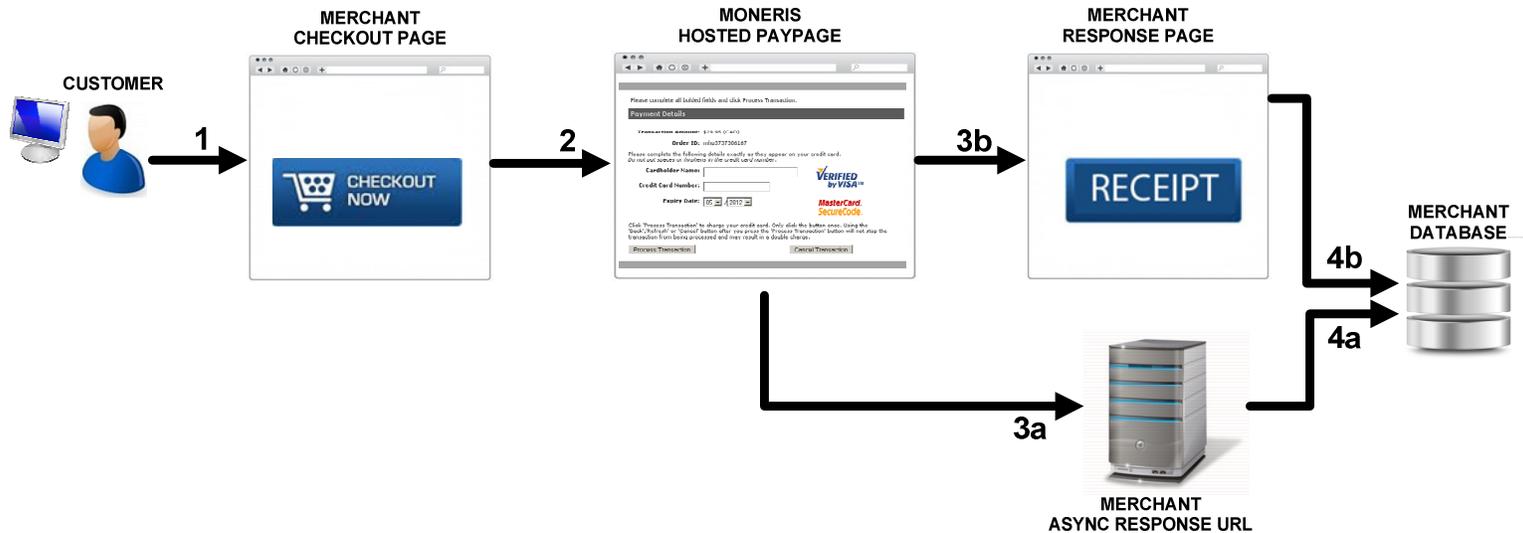
```
<FORM METHOD="POST" ACTION="https://esqa.moneris.com/HPPDP/index.php">
  <INPUT TYPE="HIDDEN" NAME="hpp_id" VALUE="4YAHJqa002">
  <INPUT TYPE="hidden" NAME="hpp_preload" >
  <INPUT TYPE="hidden" NAME="ticket" VALUE="hppEzskxQTZe8DbO80ga">
  <INPUT TYPE="SUBMIT" NAME="SUBMIT" VALUE="Click to proceed to Secure Page">
</FORM>
```

Step 4 - Normal Hosted Pay Page Process

At this point the user will reach the Hosted Pay Page where they will enter their credit card information on a secure Moneris website. After the user completes their payment they can then be sent to a merchant generated receipt page and the result of the transaction can be recorded in the merchants order management system (if applicable). If the cardholder does not proceed with charging the credit card at this point then the ticket will become invalid. A new preload request will have to be created if the cardholder wishes to checkout.



14. Hosted Pay Page Process Flow with Asynchronous Transaction Response



Outlined above is the Hosted Pay Page flow with the Asynchronous Transaction Response feature enabled.

The steps required to achieve this are as follows:

1. The customer arrives on the merchant's website (application). At this point the merchant must determine the amount of the transaction and also collect any additional data.
2. Once the cardholder is ready to pay, the merchant's checkout page will submit an HTTP form POST to the Moneris Hosted Pay Page (HPP). At this time, the customer will be redirected from the merchant's website to the Moneris Hosted Pay Page URL. For details on the HTTP form POST, please refer to section 9 of this document which outlines the mandatory and optional fields that may be sent to the Hosted Pay Page. On the Hosted Pay Page, the cardholder will fill in their secure payment details such as their card number and submit the transaction. At this time, Moneris will process the transaction and build a response.
3. 2 responses will be sent out
 - a. Moneris will perform a server to server POST of the response data to the Async Response URL provided in the Hosted Pay Page Configuration as described in section 7B of this document.
 - b. Moneris will also send an additional response back to the merchant in a POST or GET format to the Response URL provided in the Hosted Pay Page Configuration as described in section 7B of this document. The customer will also be redirected back to the merchant's website to this same Response URL.
4.
 - a. Once the merchant receives the Asynch response details they may store these details for future purposes, such as reporting, as well as use it to compare against the response received in step 3B above.
 - b. Once the merchant receives the response details they must provide a receipt to the customer and then may store these details for future purposes such as reporting or tracking. They may also at this time compare this response to the one received in step 3A above to verify the accuracy of the data.

15. Implementing the Asynchronous Transaction Response

The Asynchronous Transaction Response feature will perform a server to server POST of the response data as a secondary method of getting the response data. This does not replace the normal transaction response which will still be sent through the browser as a POST or a GET. This is a supplementary feature that can be used to verify/validate the browser response.

If this feature is enabled in the Hosted Pay Page Configuration this POST will automatically be sent back to the Async Response URL once the transaction processing is complete.

Once the merchant receives this response data, it may be used to compare to and verify the original response.

The following is a sample Asynch Response:

```
xml_response=
<response>
<response_order_id>mhp13150144643p28</response_order_id>
<bank_transaction_id>660035510010258730</bank_transaction_id>
<response_code>025</response_code>
<iso_code>00</iso_code>
<bank_approval_code>784810</bank_approval_code>
<time_stamp>14:47:45</time_stamp>
<date_stamp>2013-05-31</date_stamp>
<trans_name>purchase</trans_name>
<message>APPROVED * APPROVAL =</message>
<charge_total>1.00</charge_total>
<cardholder>test</cardholder>
<card_num>3735***5005</card_num>
<card>AX</card>
<expiry_date></expiry_date>
<result>1</result>
</response>
```

The following is a sample PHP script used to read the Asynch Transaction Response:

```
<?php
// Recieve the response from the Moneris server
$response = $_REQUEST["xml_response"];

/* remove <?xml version='1.0' standalone='yes'?>
   from the string of XML before trying to parse the XML */
$position = strpos($response, ">");
$length = strlen($response);
$response = substr ($response, $position+2, $length );

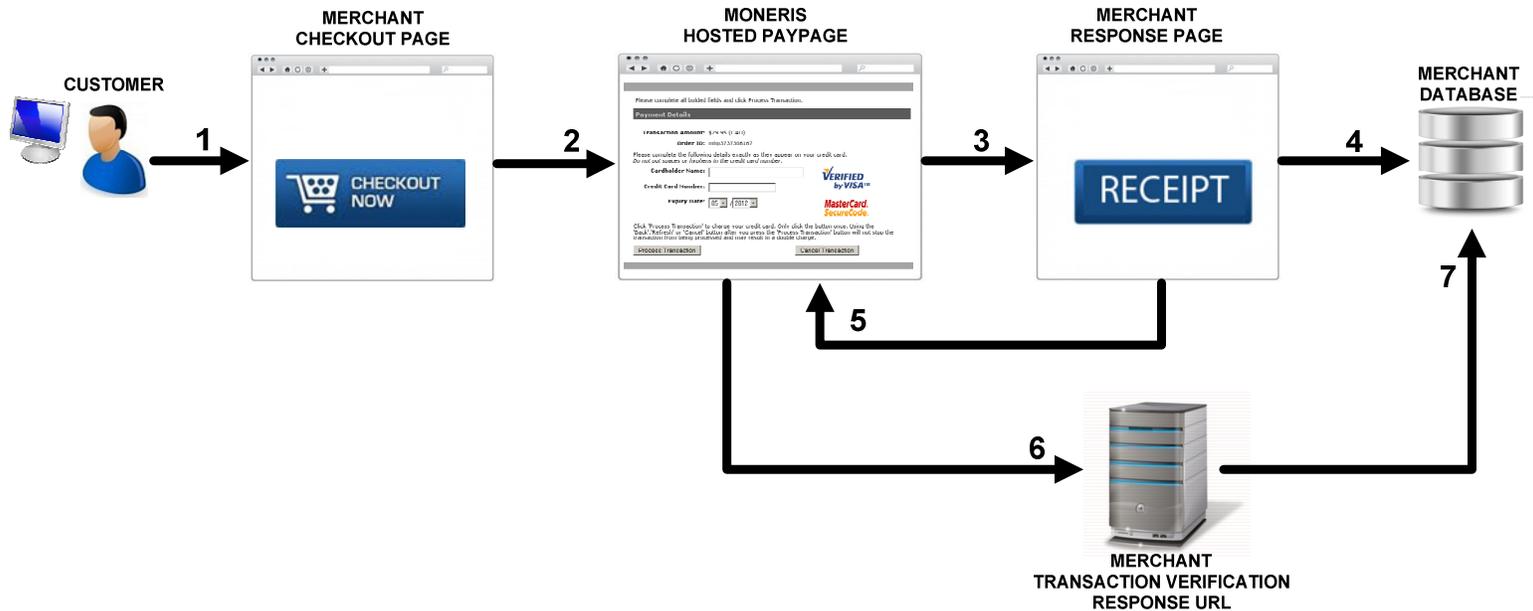
// create an array of results
$xml = simplexml_load_string($response);
foreach($xml->children() as $child)
{
    $receipt[$child->getName()] = $child;
}

// Prepare to write the results to the log file
$timestamp = date("H:i:s d/m/y", time());
$results = $timestamp . "\r\n";
$results .= "Response Order ID: " . $receipt["response_order_id"] . "\r\n";
$results .= "Bank Transaction ID: " . $receipt["bank_transaction_id"] . "\r\n";
$results .= "Response Code: " . $receipt["response_code"] . "\r\n";
$results .= "ISO Code: " . $receipt["iso_code"] . "\r\n";
$results .= "Bank Approval Code: " . $receipt["bank_approval_code"] . "\r\n";
$results .= "Time Stamp: " . $receipt["time_stamp"] . "\r\n";
$results .= "Date Stamp: " . $receipt["date_stamp"] . "\r\n";
$results .= "Transaction Name: " . $receipt["trans_name"] . "\r\n";
$results .= "Message: " . $receipt["message"] . "\r\n";
$results .= "Charge Total: " . $receipt["charge_total"] . "\r\n";
$results .= "Cardholder Name: " . $receipt["cardholder"] . "\r\n";
$results .= "Card Number: " . $receipt["card_num"] . "\r\n";
$results .= "Card Type: " . $receipt["card"] . "\r\n";
$results .= "Expiry Date(YMM): " . $receipt["expiry_date"] . "\r\n";
$results .= "Result: " . $receipt["result"] . "\r\n";

//write the results to the log file
$logFile = "async_log.txt";
$fh = fopen($logFile, 'a') or die("can't open file");
fwrite($fh, $results);
fclose($fh);
```

For further details about these response details, please refer to section 11 of this document.

16. Hosted Pay Page Process Flow with Transaction Verification



Outlined above is the Hosted Pay Page flow with the Transaction Verification feature enabled.

The steps required to achieve this are as follows:

1. The customer arrives on the merchant's website (application). At this point the merchant must determine the amount of the transaction and also collect any additional data.
2. Once the cardholder is ready to pay, the merchant's checkout page will submit an HTTP form POST to the Moneris Hosted Pay Page (HPP). At this time, the customer will be redirected from the merchant's website to the Moneris Hosted Pay Page URL. For details on the HTTP form POST, please refer to section 9 of this document which outlines the mandatory and optional fields that may be sent to the Hosted Pay Page. On the Hosted Pay Page, the cardholder will fill in their secure payment details such as their card number and submit the transaction. At this time, Moneris will process the transaction and build a response.
3. Moneris will redirect the customer back to the merchant's website as well as send the response back to the merchant in a POST or GET format to the Response URL provided in the Hosted Pay Page Configuration as described in section 7B of this document.
4. Once the merchant receives the response details they may provide a receipt to the customer at this time and then may store these details for future purposes such as reporting or tracking.
5. Once one of the fields returned in the response (step #3 above) is the transactionKey. The merchant may now send this transactionKey back to the Moneris Hosted Pay Page using another HTTP form POST. For details on the transaction verification POST, please refer to section 17 below.
6. Once the Moneris Hosted Pay Page receives this transactionKey it will look up the details we have on file for this transaction and send the amount, response code and message back to the merchant in one of the following formats: POST, GET, key/value pairs, or display XML on our server. This response will be sent to the Transaction Verification Response URL provided in the Hosted Pay Page Configuration as described in section 7B of this document. Note: the merchant may set the Response URL (step 3) and Transaction Verification URL (step 6) to be one and the same or set 2 different URLs.
7. Once the merchant receives the additional transaction verification response details, they may now display a receipt to the customer (if this was not yet done in step 4 above) and also compare this response to the one received in step 3 above to verify the accuracy of the original response data.

17. Sending a Transaction Verification Request

In order to perform a Transaction Verification it is essential that you configure the Hosted Pay Page accordingly. If the Hosted Pay Page is properly configured you will receive a variable in the GET or POST response called "transactionKey". It is advised that you log the initial transaction response and then compare the Transaction Verification response to ensure authenticity. You can use a redirect (for example, onLoad="submit") or any other

method to submit the request. We suggest automating the Transaction Verification and not using a button to submit the information. Transaction Verification can only be performed once on a given transaction, and it can only be performed within 15 minutes of the original transaction.

Transaction Verification Request

Variable Name	Type	Description
	form	https://esqa.moneris.com/HPPDP/verifyTxn.php - Development https://www3.moneris.com/HPPDP/verifyTxn.php - Production
ps_store_id	hidden	Provided by Moneris Solutions – Hosted Pay Page Configuration Tool
hpp_key	hidden	Provided by Moneris Solutions – Hosted Pay Page Configuration Tool
transactionKey	hidden	This is returned in the transaction response - refer to section 11.

Sample

Below is a sample of the transaction Verification Request:

```
<FORM NAME=VERIFY ACTION="https://esqa.moneris.com/HPPDP/verifyTxn.php">
<INPUT TYPE="HIDDEN" NAME="ps_store_id" VALUE="qampg">
<INPUT TYPE="HIDDEN" NAME="hpp_key" VALUE="hpPu7yr4Hn5k">
<INPUT TYPE="HIDDEN" NAME="transactionKey" VALUE="SDGJSGI4ujwrlsfGSGsjhlfhs5lsffs4h">
<INPUT TYPE="SUBMIT" NAME="SUBMIT" VALUE="Click to perform verification">
</FORM>
```

Once eSELECTplus receives the transaction verification request it decrypts the key, verifies and logs the request. A response is then returned with the transaction information and a status. This response is sent in the format, and to the URL, defined in the “Security Features” portion of the Hosted Pay Page configuration. Please see the table below for a list of possible Transaction Verification statuses.

Response Fields

Variable Name	Size/Type	Description
order_id	50 / an	order_id of the original transaction
response_code	3 / an	Transaction Response Code from the original transaction < 50: Transaction approved >= 50: Transaction declined NULL: Transaction was not sent for authorization * If you would like further details on the response codes that are returned please see the Response Codes document available for download at: https://developer.moneris.com
amount	n.nn (2dec)	This is the amount of the original transaction (i.e. min 0.01 & max 9999999.99)
txn_num	20 / an	Gateway Transaction identifier from the original transaction.
transactionKey	100/varchar	The transactionKey from the request
status	an	This is the value to check to see if the transaction has been properly validated. Below is a list of possible replies and their meaning. Valid-Approved : The transaction was approved and successfully validated Valid-Declined : The transaction was declined and successfully validated Invalid : No reference to this transactionKey, validation failed Invalid-ReConfirmed : An attempt has already been made with this transaction key,

validation failed

Invalid-Bad_Source : The Referring URL is not correct, validation failed

Transaction Verification Error Codes	
Code	Message / Description
991	Invalid referrer URL - <referrer url> : Referring URL does not match what is listed in the "Security Features" portion of the Hosted Pay Page configuration, validation failed. The source URL will be returned.
994	Invalid – Reconfirmed : The transaction has already been confirmed, validation failed.
995	Invalid : Not a valid confirmation request. Either the transaction doesn't exist or the request is older than 15 minutes, validation failed.

Sample

Below is a sample of the Transaction Verification Response displayed on our server in XML format

Valid Response:

```
<?xml version="1.0" standalone="yes"?>
<response>
  <order_id>ord-140708-1705</order_id>
  <response_code>27</response_code>
  <amount>15.00</amount>
  <txn_num>381038-0_7</txn_num>
  <status>Valid-Approved</status>
  <transactionKey>agtA20RwCQ60y3jA49KlaxTMYSQdS4</transactionKey>
</response>
```

Invalid Response:

```
<?xml version="1.0" standalone="yes"?>
<response>
  <order_id>ord-140708-1705</order_id>
  <response_code>994</response_code>
  <amount>null</amount>
  <txn_num>381038-0_7</txn_num>
  <status>Invalid-ReConfirmed</status>
  <transactionKey>agtA20RwCQ60y3jA49KlaxTMYSQdS4</transactionKey>
</response>
```

18. Gift Cards

Gift Card payments are now supported in the Hosted Pay Page. If your merchant account supports gift cards and gift cards are enabled in the hosted Pay Page configuration the cardholder will be given the opportunity to pay using gift cards. Each Gift Card transaction is limited to two gift cards and one other form of payment (credit card or INTERAC Online).

Due to the possible complexity of the gift card response there are only two options for handling the transaction response: eSELECTplus generates the receipt or the response is returned in XML format. If data is being returned in XML all data must be displayed on the receipt with the exception of the *txn_num* and *result*.

19. Special Error Codes

The Hosted Pay Page is designed to generate special error codes when certain data is incorrect and/or the transaction couldn't be processed. The table below contains the information regarding the error codes. Each error will be accompanied by a message describing the problem.

Special Error Codes	
Code	Message / Description
914	Transaction cancelled by cardholder – The response code indicates that the cardholder pressed the <cancel> transaction button. – This response is only returned if the enhanced cancel button functionality is enabled within the Hosted Paypage configuration.
991	Invalid referrer URL - <referrer url> – If the Hosted Pay Page solution is configured to check the referring URL and it is incorrect this error will occur. The source URL will be included in the error. Please refer to the "Security Features" portion of the Hosted Pay Page configuration for a list of all Allowed Referring URLs.
992	VBV / Secure Code authentication failed. – This error will occur if your merchant account is configured for VBV/MCSC and the cardholder failed to enter the proper PIN during the authentication process.
993	Data error - unable to store data. – This error will occur if too much request data was passed in the transaction request or if the database failed to store the request. This may occur if unsupported characters were included in one of the posted fields.
N/A	Invalid store credentials. – There is no code generated and a blank page is loaded with the above information. The ps_store_id and/or hpp_key did not match an existing store.
N/A	Card Issuer returned corrupt data. Unable to proceed with the transaction. Please return to the site where you initiated the transaction and try again. Your card has not been charged. – There is no code generated and a blank page is loaded with the above information. This error will occur if the cardholder's issuing bank did not return the correct data in the VbV/MCSC authentication process.

20. How Do I Test My Solution?

A testing environment is available for you to connect to while you are integrating your site to our payment gateway. The test environment is available 7x24; however since it is a development environment we cannot guarantee 100% availability. Also, please be aware that other merchants are using the testing environment so you may see transactions, user IDs, and Hosted Pay Page configurations that you did not create. As a courtesy to others that are testing we ask that when you are processing refunds, changing passwords and/or trying other functions that you use only the transactions/users/configurations that you created.

Using the logins in Section 7, you can create your own Hosted Pay Page Configuration ID and Token. You can use these to send transactions to our test environment and configure your Hosted Pay Page. Your configuration and Token will be valid for 30 days. You may test as often as required.

When testing you may use the following test credit card numbers with any future expiry date.

Test Card Numbers	
Card Plan	Card Number
MasterCard	5454545454545454
Visa	4242424242424242
Amex	373599005095005
Diners	36462462742008

Please note, for Gift Card test credentials please contact our Integration Support team at eselectplus@moneris.com

The test environment has been designed to replicate our production environment as closely as possible. One major difference is that we are unable to send test transactions onto the production authorization network and thus issuer

responses are simulated. Additionally, the requirement to emulate approval, decline and error situations dictates that we use certain transaction variables to initiate various response and error situations.

The test environment will approve and decline transactions based on the penny value of the amount field. For example, a transaction made for the amount of \$9.00 or \$1.00 will approve since the .00 penny value is set to approve in the test environment. Transactions in the test environment should not exceed \$10.00. This limit does not exist in the production environment. For a list of all current test environment responses for various penny values, please see the Test Environment Penny Response table as well as the Test Environment eFraud Response table, available for download at <https://developer.moneris.com>



NOTE These responses may change without notice. Moneris Solutions recommends you regularly

refer to our website to check for possible changes.

21. How Do I Activate My Store?

Once you have received your activation letter/fax go to <https://www3.moneris.com/mpg/activate> as instructed in the letter/fax. You will need to input your store ID and merchant ID then click on 'Activate'. Once this is confirmed you will need to create an administrator account that you will use to log into the Merchant Resource Centre to access and administer your eSELECTplus store. Please note that the API TOKEN that you receive during Activation is **NOT** the token that you require for the Hosted Pay Page request.

22. How Do I Configure My Store For Production?

Once you have activated you must create a Hosted Pay Page configuration. Log in to the Merchant Resource Centre (<https://www3.moneris.com/mpg>) using the Username and Password that was created during Activation. Click ADMIN menu item and then HOSTED PAGE CONFIG. You will then need to click on "Generate a New Configuration". You will now be provided with your Configuration ID (ps_store_id) and Hosted Pay Page Token (hpp_key).

Once you have completed your testing you are ready to point your store to the production host. You will need to change the <FORM METHOD="POST" ACTION=<https://esqa.moneris.com/HPPDP/index.php>> to contain the production URL: <FORM METHOD="POST" ACTION=<https://www3.moneris.com/HPPDP/index.php>>. You will also need to change the ps_store_id and hpp_key to reflect your production store configuration.

Once you are in production you will access the Merchant Resource Centre at <https://www3.moneris.com/mpg>. You can use the store administrator ID you created during the activation process and then create additional users as needed.

23. How Do I Get Help?

If you require technical assistance while integrating your store, please contact the eSELECTplus Support Team:

For technical support:

Phone: 1-866-319-7450 (Technical Difficulties)

Email: eselectplus@moneris.com

For integration support (8:00 am – 8:00 pm EST):

Phone: 1-866-562-4354

Email: eproducts@moneris.com

When sending an email support request please be sure to include your name and phone number, a clear description of the problem as well as the integration method you are using (Hosted Pay Page). **For security reasons, please do not send us your ps_store_id combined with your hpp_key, or your merchant number and device number in the same email.**

24. Appendix A: Example

Transactions Requests

The example below will send both shipping and billing address information as well as item information, and initiate a recurring charge.

```
<FORM ACTION="https://esqa.moneris.com/HPPDP/index.php" method=post>
<!-- Store Settings-->
<INPUT TYPE="HIDDEN" NAME="ps_store_id" VALUE="qampg">
<INPUT TYPE="HIDDEN" NAME="hpp_key" VALUE="hpPu7yr4Hn5k">

<!------- DEFINE CHARGE TOTAL HERE ---->
Amount:<INPUT TYPE="text" NAME="charge_total" VALUE="300.00"><br><br>

<!-- Unique Order ID -->
<INPUT TYPE="hidden" NAME="order_id" VALUE="hpp_mr_test_1">

<!-- Additional Optional Details -->
<input type="hidden" name="cust_id" value="customer num">
<input type="hidden" name="email" value="">
<input type="hidden" name="note" value="these are special instructions">

<!-- Item Information -->
<input type="hidden" name="quantity1" value="3">
<input type="hidden" name="description1" value="qunat 3">
<input type="hidden" name="id1" value="skul23">
<input type="hidden" name="price1" value="4.00">
<input type="hidden" name="subtotal1" value="12.00">

<input type="hidden" name="quantity2" value="2">
<input type="hidden" name="description2" value="qunat 2">
<input type="hidden" name="id2" value="2skul23">
<input type="hidden" name="price2" value="24.00">
<input type="hidden" name="subtotal2" value="212.00">

<input type="hidden" name="gst" value="3.03">
<input type="hidden" name="shipping_cost" value="4.03">

<!-- rvar Information -->
<input type="hidden" name="rvar1" value="1_rvar">
<input type="hidden" name="rvar2" value="2_rvar">
<input type="hidden" name="rvar3" value="3_rvar">
<input type="hidden" name="rvar4" value="4_rvar">

<!-- Shipping information -->
<input type="hidden" name="ship_first_name" value="sfn">
<input type="hidden" name="ship_last_name" value="sln">
<input type="hidden" name="ship_company_name" value="scn">
<input type="hidden" name="ship_address_one" value="sao">
<input type="hidden" name="ship_city" value="sc">
<input type="hidden" name="ship_state_or_province" value="ssop">
<input type="hidden" name="ship_postal_code" value="spc">
<input type="hidden" name="ship_country" value="scount">
<input type="hidden" name="ship_phone" value="sp">
<input type="hidden" name="ship_fax" value="sf">
```

```
<!-- Billing Information -->
<input type="hidden" name="bill_first_name" value="bfn">
<input type="hidden" name="bill_last_name" value="bln">
<input type="hidden" name="bill_company_name" value="bcn">
<input type="hidden" name="bill_address_one" value="bao">
<input type="hidden" name="bill_city" value="bc">
<input type="hidden" name="bill_state_or_province" value="bsop">
<input type="hidden" name="bill_postal_code" value="bpc">
<input type="hidden" name="bill_country" value="bcount">
<input type="hidden" name="bill_phone" value="bp">
<input type="hidden" name="bill_fax" value="bf">

<!-- Recurring Information -->
<input type="hidden" name="doRecur" value="1">
<input type="hidden" name="recurUnit" value="day">
<input type="hidden" name="recurStartDate" value="2006/06/01">
<input type="hidden" name="recurNum" value="99">
<input type="hidden" name="recurStartNow" value="true">
<input type="hidden" name="recurPeriod" value="4">
<input type="hidden" name="recurAmount" value="4.00">

<INPUT TYPE="SUBMIT" NAME="SUBMIT" VALUE="Click to proceed to Secure Page">
</FORM>
```

25. Appendix B. Card Validation Digits (CVD) and Address Verification Service (AVS)

Card Validation Digits (CVD)

The Card Validation Digits (CVD) value refers to the numbers appearing on the back of the credit card which are not imprinted on the front. The exception to this is with American Express cards where this value is indeed printed on the front

Address Verification Service (AVS)

The Address Verification Service (AVS) value refers to the cardholder's street number, street name and zip/postal code as it would appear on their statement.

Additional Information for CVD and AVS

The responses that are received from CVD and AVS verifications are intended to provide added security and fraud prevention, but the response itself will not affect the issuer's approval of a transaction. Upon receiving a response, the choice to proceed with a transaction is left entirely to the merchant.

Please note that all responses coming back from these verification methods are not direct indicators of whether a merchant should complete any particular transaction. The responses should not be used as a strict guideline of which transaction will approve or decline.



Please note that CVD verification is only applicable towards Visa, MasterCard and American Express transactions.

Also, please note that AVS verification is only applicable towards Visa, MasterCard, Discover and American Express transactions. This verification method is not applicable towards any other card type.

***For additional information on how to handle these responses, please refer to the eFraud (CVD & AVS) Result Codes document which is available at <https://developer.moneris.com>**

26. Appendix C. XML POST Response for Financial Transaction

The XML is returned in a field called `xml_response`. A sample of the XML is below. Fields in blue are returned optionally or based on the transaction type performed. The XML should be parsed dynamically to ensure that if and when fields are added in the future the transaction responses are handled properly. Field definitions are the same as indicated in the standard response format tables. Gift card response fields are defined below. The `gift_card` tag may appear once or twice depending on the number of gift cards used during the transaction. The `item` tag will appear for every item that was posted in the request with a quantity greater than 0.


NOTE

The XML may not be returned formatted in the manner below. It may be returned as a single line or with line breaks. Your XML parser should be able to handle these variations.

```
<?xml version='1.0' standalone='yes'?>
<response>
  <response_order_id>mhp1573006623</response_order_id>
  <bank_transaction_id>660035520011120030</bank_transaction_id>
  <response_code>027</response_code>
  <iso_code>01</iso_code>
  <bank_approval_code>608681</bank_approval_code>
  <time_stamp>18:53:27</time_stamp>
  <date_stamp>2008-07-10</date_stamp>
  <trans_name>purchase</trans_name>
  <message>APPROVED * =</message>
  <charge_total>1.00</charge_total>
  <cardholder>Bill Smith</cardholder>
  <card_num>4510***5010</card_num>
  <card>V</card>
  <expiry_date>0807</expiry_date>
  <result>1</result>
  <eci>7</eci>
  <txn_num>829-0_22</txn_num>
  <rvar1>1_rvar</rvar1>
  <rvar2>2_rvar</rvar2>
  <transactionKey>uJv2RGGasX4Kd3Tlz3eujRAY5wUCd1</transactionKey>
  <recur_result> </recur_result>
  <gift_card>
    <order_no>mhp1573006623_g1</order_no>
    <txn_num>9041-1215730102475-00035540_21</txn_num>
    <response_code>000</response_code>
    <ref_num>37286815</ref_num>
    <terminal_id>00035540</terminal_id>
    <txn_type>purchase</txn_type>
    <card_num>0211***0222</card_num>
    <card_desc>Gift Fixed Reloadabl</card_desc>
    <date_time>Jul 10 2008 06:53PM</date_time>
    <gift_charge_total>99.50</gift_charge_total>
    <rem_balance>0.00</rem_balance>
    <display_text>Approved</display_text>
    <receipt_text>En
      □NHLJ: Jul 10, 2008
      □NHC: 07/10/2008
      □NHRJ: 10-07-2008
    </receipt_text>
    <voucher_text> </voucher_text>
    <result>1</result>
  </gift_card>
  <item>
    <quantity>3</quantity>
    <description>qunat 3</description>
    <id>skul23</id>
```

DHLJ: 10 Jul 2008D

DHC: Jul 10, 2008

DHRJ: End Of Text

```
        <price>4.00</price>
    </item>
    <item>
        <quantity>2</quantity>
        <description>qunat 2</description>
        <id>2skul23</id>
        <price>24.00</price>
    </item>
    <item_misc>
        <shipping_cost>4.03</shipping_cost>
        <hst></hst>
        <pst></pst>
        <gst>3.03</gst>
    </item_misc>
    <shipping>
        <ship_first_name>sfm</ship_first_name>
        <ship_last_name>sln</ship_last_name>
        <ship_company_name>scn</ship_company_name>
        <ship_address_one>sao</ship_address_one>
        <ship_state_or_province>ssop</ship_state_or_province>
        <ship_postal_code>spc</ship_postal_code>
        <ship_country>scount</ship_country>
        <ship_phone>sp</ship_phone>
        <ship_fax>sf</ship_fax>
    </shipping>
    <billing>
        <bill_first_name>bfm</bill_first_name>
        <bill_last_name>bln</bill_last_name>
        <bill_company_name>bcn</bill_company_name>
        <bill_address_one>bao</bill_address_one>
        <bill_state_or_province>bsop</bill_state_or_province>
        <bill_postal_code>bpc</bill_postal_code>
        <bill_country>bcount</bill_country>
        <bill_phone></bill_phone>
        <bill_fax></bill_fax>
    </billing>
    <od_other>
        <email>bill.smith@example.com</email>
        <cust_id>customer num</cust_id>
        <note>these are special instructions</note>
    </od_other>
</response>
```

27. Appendix D. Sample Receipt

The sample below displays the layout of the response receipt. This receipt is only displayed if the *Response Method* is set to "eSELECTplus will generate a receipt" in the Hosted Pay Page configuration. The sections and sub sections displayed, as well as the color and style of the receipt, are determined by the settings chosen in the Hosted Pay Page configuration. Please refer to section 7 - The Merchant Resource Centre Hosted Pay Page Configuration Tool for further information.

GIFT CARD INTEGRATION

TRANSACTION APPROVED - THANK YOU

Please print this page and keep it as your transaction receipt.

Payment Details

Transaction Type: PURCHASE

Date / Time: 2008-07-15 17:17:28

Transaction Amount: \$15.00 CAD

Order ID: ord-150708-1657

Cardholder: Bob Smith

Card Number: 4242***4242

Card Type: V

Resp Code - Message: 027 - APPROVED * =

Reference Number: 660060730010121030

Auth Code: 975620

Refund Policy: No refunds, only exchanges within 0.5 day

Item Details

Description	Product Code	Quantity	Price
Member fee	1	1	\$15.00
			Shipping: \$0.00
			GST: \$0.00
			PST: \$0.00
			HST: \$0.00
			Total (CAD): \$15.00

Customer Details

Customer ID: My personal Customer ID

Email Address: proper@email.com

Note: Please deliver to back door

Billing Address

First Name: Mary

Last Name: Smith

Company: Moneris

Address: 1 Lakeshore Blvd

City: Brampton

Province/State: ON

Country: Canada

Postal Code: L1L1L1

Phone: 905-555-3333

Fax: 905-555-4444

Shipping Address

First Name: Bob

Last Name: Smith

Company: Moneris

Address: 101 Main St

City: Springfield

Province/State: NT

Country: Canada

Postal Code: N1N1N1

Phone: 866-555-1111

Fax: 866-555-2222

Continue

28. Appendix E: Sample Hosted Pay Page

The sample below displays the layout of the Hosted Pay Page. The sections and sub sections displayed, as well as the color and style of the page, are determined by the settings chosen in the Hosted Pay Page Configuration. Please refer to section 7- The Merchant Resource Centre Hosted Pay Page Configuration Tool for further information.

QA Merchant 5 - eFraud

Item Details			
Description	Product Code	Quantity	Price
Member Fee	R	1	\$4.00
			GST: \$0.00
			PST: \$0.00
			HST: \$0.52
Total (CAD):			\$4.52

Customer Details	
Customer ID: A customer identifier	
Email Address: proper@email.com	
Note:	
Billing Address	Shipping Address
First Name: Mary	First Name: Bob
Last Name: Smith	Last Name: Smith
Company: Moneris	Company: Moneris
Address: 1 Lakeshore Blvd	Address: 1 Lakeshore Blvd
City: Brampton	City: Brampton
Prov/State: ON	Prov/State: ON
Country: CA	Country: CA
Postal Code: L1L1L1	Postal Code: L1L1L1
Phone: 905-555-3333	Phone: 866-555-1111
Fax: 905-555-4444	Fax: 866-555-2222

Cardholder Details
Please enter the following address exactly as it appears on your credit card statement.
PO Box: <input type="checkbox"/>
Street Number: <input type="text"/>
Street Name: <input type="text"/>
Postal/Zip Code: <input type="text"/>

Payment Details	
Transaction Amount: \$4.52 (CAD)	     
Order ID: mhp13177120341p55	 
Please complete the following details exactly as they appear on your credit card. Do not put spaces or hyphens in the credit card number.	
Cardholder Name: <input type="text"/>	
Credit Card Number: <input type="text"/>	
Expiry Date: 06 / 2013	
Card Security Code: <input type="text"/>	

Click 'Process Transaction' to charge your credit card. Only click the button once. Using the 'Back', 'Refresh' or 'Cancel' button after you press the 'Process Transaction' button will not stop the transaction from being processed and may result in a double charge.

Process Transaction

Cancel Transaction

29. Appendix F. CAVV Result Code

The Cardholder Authentication Verification Value (CAVV) is a value that allows VisaNet to validate the integrity of the VbV transaction data. These values are passed back from the issuer to the merchant after the VbV/SecureCode authentication has taken place.

The following table describes the contents of the CAVV data response and what it means to the merchant.

Table of CAVV result codes		
Result Code	Message	What this means to you as a merchant...
0	CAVV authentication results invalid.	For this transaction you may not receive protection from chargebacks as a result of using VbV as the CAVV was considered invalid at the time the financial transaction was processed. Please check that you are following the VbV process correctly and passing the correct data in our transactions.
1	CAVV failed validation; authentication	Provided that you have implemented the VbV process correctly the liability for this transaction should remain with the Issuer for chargeback reason codes covered by Verified by Visa.
2	CAVV passed validation; authentication	The CAVV was confirmed as part of the financial transaction. This transaction is a fully authenticated VbV transaction (ECI 5)
3	CAVV passed validation; attempt	The CAVV was confirmed as part of the financial transaction. This transaction is an attempted VbV transaction (ECI 6)
4	CAVV failed validation; attempt	Provided that you have implemented the VbV process correctly the liability for this transaction should remain with the Issuer for chargeback reason codes covered by Verified by Visa.
7	CAVV failed validation; attempt (US issued cards only)	Please check that you are following the VbV process correctly and passing the correct data in our transactions. Provided that you have implemented the VbV process correctly the liability for this transaction should be the same as an attempted transaction (ECI 6)
8	CAVV passed validation; attempt (US issued cards only)	The CAVV was confirmed as part of the financial transaction. This transaction is an attempted VbV transaction (ECI 6)
9	CAVV failed validation; attempt (US issued cards only)	Please check that you are following the VbV process correctly and passing the correct data in our transactions. Provided that you have implemented the VbV process correctly the liability for this transaction should be the same as an attempted transaction (ECI 6)
A	CAVV passed validation; attempt (US issued cards only)	The CAVV was confirmed as part of the financial transaction. This transaction is an attempted VbV transaction (ECI 6)
B	CAVV passed validation	The CAVV was confirmed as part of the financial transaction. However, this transaction does not qualify for the liability shift. Treat this transaction the same as an ECI 7.

eSELECTplus™

Copyright Notice

Copyright © 2013 Moneris Solutions, 3300 Bloor Street West, Toronto, Ontario, M8X 2X2

All Rights Reserved. This manual shall not wholly or in part, in any form or by any means, electronic, mechanical, including photocopying, be reproduced or transmitted without the authorized, written consent of Moneris Solutions.

This document has been produced as a reference guide to assist Moneris client's hereafter referred to as merchants. Every effort has been made to the make the information in this reference guide as accurate as possible. The authors of Moneris Solutions shall have neither liability nor responsibility to any person or entity with respect to any loss or damage in connection with or arising from the information contained in this reference guide.

Trademarks

Moneris and the Moneris Solutions logo are registered trademarks of Moneris Solutions Corporation.

Any software, hardware and or technology products named in this document are claimed as trademarks or registered trademarks of their respective companies.

Printed in Canada.

10 9 8 7 6 5 4 3 2 1