How to Use the Central Credit Card Payment Site (CPS)

Overview

The Central Payment Site is a toolkit that allows WSU departments to develop secure e-commerce applications that meet the PCI (Payment Card Industry) Data Security Standards which were mandated in July, 2005. The Central Payment Site toolkit (CPS) consists of web services and a web page. The web services are used to insert and read information to/from a secure database. The web page is a form that prompts the customer for the credit card type, credit card number, expiration date and CVV (Customer Validation Verification code). The web page is displayed from a secure central server hosted Cybersource – not the departmental application. When it is time to collect a payment from a customer, the departmental application re-directs to the CPS web page. After collecting the payment from the customer, the CPS web page re-directs back to the departmental application. The departmental application then does its backend processing such as displaying a receipt to the customer, crediting the customer's account, initiating shipment processes, etc. Because the departmental application does not 'input, process or store' the credit card number, it is by definition not subject to the PCI standards. To get a feel for the flow of the application, go through the demonstration at: https://dev-webapps.wsu.edu/ais/eCommerce/eCommerceTest/CentralPaymentSiteDemo/DeptRequsetPaymentAuthorization.aspx

Getting Started

The business aspects of your e-commerce application are well documented at http://ecommerce.wsu.edu. There you will find details on applying for merchant numbers, having a new t_type created, access request forms for Cybersource and the EAMS query system.

Merchant Numbers

Before you can begin processing credit cards you will need a merchant number. A merchant number is an account with our merchant bank – Bank of America. WSU requests merchant numbers as needed for new applications. The primary significance of a merchant number is that it determines the description of the charge that will occur on the customer's bank statement. The description can only be 13 characters in length and is followed by a 10 digit contact phone number (e.g. "WSU Go Cougs 509-335-1234"). The contact phone number should be for a person that can answer questions about the charge (e.g. refund or return policy, status of the order, etc.). If your department currently has Internet-based e-commerce applications, it can likely use the merchant number being used by your existing applications. Contact Stacy Combs (combssc@wsu.edu, 509-335-2039) to see if you can use an existing merchant number or a new one needs to be applied for.

T-Types

When a credit card transaction from your application is approved, your department's budget-project in the WSU accounting system gets credit for the transaction. Your application however does not reference the budget-project directly however. Instead, it references a code (t-type) from the One Step Point of Sale system. The t-type is linked to a specific budget-project and source and sub-source. Controller's office staff is responsible for setting up t-types. A departmental e-commerce application may use more than one t_type if they want revenue to be credited to more than one account. E.g. a conferencing application might want the revenue for each conference to be credited to a separate account. To request a new t_type, see the form on this Sharepoint site.

IP Security

The web services are secured by IP address. Send an email bsgprog@lists.wsu.edu with the IP address of the test web server. Depending on your development environment, you might also need the IP address of your workstation to be added to the list of permitted IP addresses.

Web Services

How to Use the Central Credit Card Payment Site (CPS)

The web service definitions can be found at:
https://test-ewebservice.wsu.edu/CentralPaymentSite_WS/service.asmx -- Test
https://ewebservice.wsu.edu/CentralPaymentSite_WS/service.asmx -- Production

Developing the Application

For most credit card applications the procedure to charge a credit card is a simple three step process:
1) Perform a web service call to 'set-up' the transaction. The web service call supplies the web page with necessary information including, merchant name, amount of the payment, t_type, primary-id, etc.
2) Re-direct to the CPS web page;
3) Perform a web service call to get the results of the credit card transaction. E.g. approval code, authorization number, masked credit card number, etc. Below is the process flow of an authorization-capture transaction in more detail:

Process Flow for Auth-CapturewithAddress

Web Server	Action	Notes
Departmental Application Web Server	1.Call AuthCapRequestwithAddress web service to set-up the credit card payment.	
	2. Redirect to the CPS web page.	
Central Payment Site Web Server	3. Re-directs to Cybersource Hosted Order page.	
	4. Input credit card data and process request.	
	5. Update request with results.	
	6. Postback results to the Postback URL supplied by the department (to update backend database).	
	7. Redirect to ReturnURL supplied by the application (to display customer receipt).	
Departmental Application Web Server	8. Call AuthCapResponse web service from the Postback URL application.	
	9. Call AuthCapResponse web service from the ReturnURL application and display approved/declined message to customer.	

How to Use the Central Credit Card Payment Site (CPS)

Process Flow for Authorization and Capture

NOTE: The CaptureRequest method requires that a security key be configured for your merchant Account. If your merchant account has never used the CaptureRequest method, contact Bsgprog@ Lists.wsu.edu to request that a Capture security key be configured for your merchant account.

Web Server	**Action**	**Notes**
Departmental Application Web Server	1.Call AuthRequestwithAddress web service to set-up the credit card payment.	
	2. Redirect to the CPS web page.	
Central Payment Site Web Server	3. Input credit card data and process request using CPM server.	
	4. Update request with results.	
	5. Redirect to ReturnURL supplied by the application.	
Departmental Application Web Server	6.Call AuthCapResponse web service to find out the results of the credit card authorization.	
	7. Call CaptureRequest to complete the transaction. NOTE: CaptureRequest can only be called once for each AuthRequest.	

Definitions

The table below lists in alphabetical order the name, type and definition of the input and returned parameters from the web service calls

Field Name	**Type**	**Definition**
ApplicationIDPrimary	String	This is an application defined field that is used to help track and identify the transaction in the case of refunds, problems or reporting. It could be a WSU number, customer number, invoice number, conference id, etc. Maximum length of 50. Stored in the CPM database
ApplicationIDSecondary	String	Additional descriptive information about the payment transaction. It often supports the primary id. Examples of secondary Ids are name, phone number, city and order number. Maximum length of 50. Stored in the CPM database
ApplicationStateData	String	Optional data area that can be used to maintain application state. Maximum length of 200.
ApprovalCode	String	Approval code sent by the credit card processor. Maximum length of 9.
Approved_Transactions_Count	Int	The number of approved transactions in the CPM database that meet all of the search criteria supplied to the CheckCPMDatabase service.
AuthorizationAmount	Decimal	Amount to be charged.

AuthorizationAttemptLimit	Int	Number of times the central credit card authorization web page will allow a user to enter a credit card number and attempt an authorization before declining the request and reporting the failure back to application. Values: 1, 2 or 3
AuthorizationType	String	Contains the type of credit card authorization requested: Values: "AUTH" and "AUTHCAP".
BeginDateTime	DateTime	Used in the CheckCPMDatabase service, this is the transaction search beginning date-time.
BillingAddress	String	The street address associated with the customers statement. Maximum length of 60.
BillingCity	String	The city associated with the customer's statement. Maximum of 50.
BillingZipCode	String	The Zip or Postal Code (Canadian or US) associated with the customer's statement. Maximum of 10.
BillingCountry	String	The state or province associated with the customer's statement. Either 'US' or 'CA'.
BillingState	String	Two character state or province abbreviation associated with the customer's statement.
EndDateTime	DateTime	Used in the CheckCPMDatabase service, this is the transaction search ending date-time.
CaptureAmount	Decimal	Amount to be captured from a previously successful AuthRequest.
Check_CPM_Return_Code	Int	The return code issued by the web service. A '0' indicates that the call was successful – no errors encountered. A non-zero return code indicates an error condition. A description of the problem is in the Check_CPM_Return_Message.
Check_CPM_Return_Message	String	A description of the Check_CPM_Return_Code.
CPMReturnCode	Integer	This contains the results of the attempted credit card charge: 0 = Your charge was successful, 1 = Card was declined (details in CPMReturnMessage), 2 = A processing error has occurred (details in CPMReturnMessage), 3 = bad input parameter (name of field in CPMReturnMessage). Any other return code is a system error and a description will be put in the CPMReturnMessage.
CPMReturnMessage	String	Textual explanation of CPMReturnCode.
CPMSequenceNum	String	This is a unique key that identifies the credit card transaction data. This number is assigned by the CPM server and stored in the CPM database. Many e-commerce applications store this number so they can reference the transaction for the purpose of refunds or troubleshooting. Maximum length of 15.
CreditCardType	String	Type of credit card i.e. "Visa" "MasterCard". Maximum length of 20.
EmailAddressDeptContact	String	An e-mail address supplied by the application. This address will be used to e-mail error notifications to a departmental contact that occur in the process of authorizing a transaction. Maximum length of 128.
MaskedCreditCardNumber	String	The masked credit card number with only the last 4 digits shown – "xxxxxxxxxxxx1234'. Maximum length of 16.

MerchantID	String	This is an alphanumeric mnemonic that points to a Bank of America merchant number. The MerchantId is assigned by the e-commerce support person in IT, currently, John Chapman. Maximum length of 32.
OneStepTranType	String	Transaction type used in One-Step POS system. The OneStepTranType tells the WSU accounting system which budgetary account (budget-project) gets credited with the revenue from the credit card transaction. Transaction types or “t_types” are created in the One-Step POS system by Controller’s Office staff on request. A departmental e-commerce application may use more than one t_type if they want revenue to be credited to more than one account. E.g. a conferencing application might want the revenue for each conference to be credited to a separate account. To request a new t_type, see the form on this Sharepoint site. Maximum length of 8.
PostbackURL	String	Application URL that the Cybersource Hosted Order Page will Postback to after the credit card authorization. Maximum length of 512.
ResponseReturnCode	Int	Return code from the web service call. Return code of 0 = success, anything else is an error.
ResponseReturnMessage	String	
RequestGUID	String	This is a unique Id that is assigned by and returned from the AuthCapRequest or AuthRequest web services. It is a key to the record in the database that contains the detailed information about the credit card authorization attempt. The RequestGUID is passed to the CPS web page and the AuthCapResponse web service by the application. Maximum length of 36.
ReturnCode	Int	Return code 0 = success ….
ReturnMessage	String	Error message. Maximum length of 80.
ReturnURL	String	Application URL that the central credit card web page will redirect to after processing the request. Maximum length of 512.
StyleSheetKey	String	This is the WSU Template Designer Key which is used to create designs that adhere to the WSU Web Identity Guidelines. This enables you to customize the CPS web page so that it integrates well with your application. See http://designer.wsu.edu/template/ for more information. Maximum length of 16.
WebPageURLAndGUID	String	This is the URL of the CPS web page. It is returned to the application by the AuthCapRequest and AuthRequest web services. This is the URL that the application should re-direct to at the time when you want to collect the payment. Your GUID is appended to the Web Page URL.

How to Use the Central Credit Card Payment Site (CPS)

Required and Optional Parameters
The table below documents the parameters for each web service call and indicates which input parameters are required and which are optional:

Name / Description	Input	Returned
AuthCapRequestWithAddress Create an authorization/capture request.	**Required:** MerchantID AuthorizationAmount OneStepTranType ApplicationIDPrimary ReturnURL PostBackURL AuthorizationAttemptLimit **Optional:** (may be blank) ApplicationIDSecondary ApplicationStateData StyleSheetKey EmailAddressDeptContact BillingAddress BillingCity BillingState BillingZipCode BillingCountry	RequestGUID RequestReturnCode RequestReturnMessage WebPageURLAndGUID
AuthRequestWithAddress Create an authorization request.	**Required:** MerchantID AuthorizationAmount OneStepTranType ApplicationIDPrimary ReturnURL PostBackURL AuthorizationAttemptLimit **Optional:** (may be blank) ApplicationIDSecondary ApplicationStateData StyleSheetKey EmailAddressDeptContact BillingAddress BillingCity BillingState BillingZipCode BillingCountry	RequestGUID RequestReturnCode RequestReturnMessage WebPageURLAndGUID

AuthCapResponse Get the results of an AuthCapRequest. This web service is called to return the results of the credit card transaction to the application.	**Required:** GUID	ResponseReturnCode ResponseReturnMessage AuthorizationType MerchantID ResponseGUID AuthorizationAmount OneStepTranType ApplicationIDPrimary ApplicationIDSecondary ApplicationStateData CPMReturnCode CPMReturnMessage ApprovalCode CPMSequenceNum AuthorizationAttemptLimit StyleSheetKey CreditCardType MaskedCreditCardNumber
CaptureRequest Create a capture request based on an existing authorization.	**Required:** GUID CaptureAmount OneStepTranType	ResponseReturnCode ResponseReturnMessage ApplicationStateData CPMReturnCode CPMReturnMessage ApprovalCode CPMSequenceNum CreditCardType MaskedCreditCardNumber

Testing the Application

The following test credit card numbers are available for testing:

Visa:	4111111111111111
	4234567890123456
MasterCard:	5424000000000015

Send any information you want to in your test cases, using the above credit card numbers. If you encounter UNEXPECTED errors, or unhandled exceptions, please forward a screen shot of the error, with the data you sent and the date and time the error occurred to: bsgprog@lists.wsu.edu

Moving the Application into Production

After you are comfortable with your test results and are ready for your application to go-live, send an email to bsgprog@lists.wsu.edu with the IP of your production server. Request that it be permitted to the production web services. After your IP has been permitted, change your application to reference the production web services at: https://ewebservice.wsu.edu/CentralPaymentSite_WS/service.asmx

How to Use the Central Credit Card Payment Site (CPS)

Code Snippets

Sample VB code to set-up the call to the AuthCapRequestwithAddress service and re-direct to the web page

```
'Instantiate the return parameter structure
Dim rr As New edu.wsu.test_ewebservice.Request_Return_Parms
'Instantiate the web service
Dim myCCAuthRequest As New edu.wsu.test_ewebservice.CCAuthRequestwithAddress
Dim MerchantID As String = "XXXXXXXX"
Dim AuthorizationAmount As Decimal
Dim OneStepTranType As String = "XXXXXXXX"
Dim ApplicationIDPrimary As String
Dim ReturnURL As String = _
    https://YourDnsName.wsu.edu/AuthCapResults.aspx
Dim PostbackURL As String =
    https://YourDNSName.wsu.edu/app.aspx --app that receives postback from cyber
Dim AuthorizationAttemptLimit As String = "2"
Dim ApplicationIDSecondary As String = ""
Dim ApplicationStateData As String = ""
Dim StyleSheetKey As String = ""
Dim EmailAddressDeptContact As String = "TechSupportPerson@VetMed.WSU.EDU"
Dim myCCAuthRequest As New edu.wsu.test_ewebservice.CCAuthRequest

AuthorizationAmount = . . .
ApplicationIDPrimary = . . .
rr = myCCAuthRequest.AuthCapRequestWithAddress(MerchantID, _
                                    AuthorizationAmount, _
                                    OneStepTranType, _
                                    ApplicationIDPrimary, _
                                    ReturnURL, _
                                    AuthorizationAttemptLimit, _
                                    ApplicationIDSecondary, _
                                    ApplicationStateData, _
                                    StyleSheetKey, _
                                    EmailAddressDeptContact,_
                                    BillingAddress, _
                                    BillingCity, _
                                    BillingState, _
                                    BillingZipCode, _
                                    BillingCountry, _
                                    PostBackURL)

If rr.RequestReturnCode = 0 Then
    'The call to the web service was successful, re-direct to the web page . . .
    Response.Redirect(rr.WebPageURLAndGUID)
Else
    'Process error . . .'
End If
```

* * * * * * * * * * * * * * * * * * *

Sample code to check the results of the credit card transaction from re-direction

```
Dim myCCAuthRequest As New edu.wsu.test_ewebservice.AuthCapResponse
Dim rr As New edu.wsu.test_ewebservice.Response_Return_Parms
Dim ReturnedGUID As String
```

How to Use the Central Credit Card Payment Site (CPS)

```
'Extract the GUID passed from the Central Payment Site web page
ReturnedGUID = Request.QueryString("GUID")

'Read the results of the credit card transaction

rr = myCCAuthRequest.AuthCapResponse(ReturnedGUID)


If rr.ResponseReturnCode = 0 Then
'Call to the web service was successful
   If rr.CPMReturnCode = 0 Then
   'The credit card was approved, display receipt, perform any
     backend/inventory processing
   Else
      If rr.CPMReturnCode = 1 Then
         'The credit card was declined by the customer’s bank.
      Else
         'System or internal error occurred . . .
      End-if
   End-if
ElseIf rr.ResponseReturnCode = 2   ‘this guid already closed
    ‘This is the normal case—your transaction was closed by the postback routine
    ‘Display receipt to customer
Else
   'The call to the web service failed . . .
End-if
```

Sample code to check the results of the credit card transaction from Cybersource postback

```
Dim guidStr As String
Dim returnCode as String
Dim returnMsg as String

If Request.Form.Get("GUID") IsNot Nothing Then

    guidStr = Request.Form.Get("GUID").ToString.Trim
    If Request.Form.Get("RETURNCODE") IsNot Nothing Then
        ReturnCode = Request.Form.Get("RETURNCODE").ToString.Trim
    End If
    If Request.Form.Get("MESSAGE") IsNot Nothing Then
        ReturnMsg = Request.Form.Get("MESSAGE")
    End If

End If

If (guidStr isNot nothing) AndAlso (returnCode == 0) then
    update your database, credit the payment.
    If your update success then
        Call AuthCapResponse(guidStr) to close this transaction
     Else
         Do whatever you need.
    EndIf
End if
```