

CCAvenue Integration Document



Version 2.4

Phone: +91 022-67425555

January 2014

INDEX

<u>CCAvenue Integration Document</u>	3
<u>Testing and Production Environment</u>	4
<u>Integration Methods</u>	5
<u>Processing orders using CCAvenue billing page</u>	6
• <u>Basic steps involved in integration with the CCAvenue billing page</u>	7
<u>Processing orders using CCAvenue iFrame Checkout</u>	12
• <u>Basic steps involved in integration CCAvenue iFrame into Checkout page</u>	13
<u>Processing orders using custom checkout form</u>	18
• <u>Basic steps involved in fetching payment options to create custom checkout form</u>	19
<u>Vault feature for storing card</u>	25
<u>Processing orders using CCAvenue shopping cart</u>	26
• <u>Basic steps involved in integration with the CCAvenue shopping cart</u>	27
<u>Contact Details</u>	34

CCAvenue Integration Document

Introduction

CCAvenue payment integration kit allows merchants to instantly collect payments from their users using various payment modes like credit cards, debit cards, cash cards, net banking etc.

The CCAvenue payment integration supports a seamless payment experience on your platform, while protecting your application from payment frauds and complexity related to various regulations.

Testing and Production Environment

CCAvenue test and production environments are separate.

Merchants need an active CCAvenue account to use the test environment and production environment. Merchants will have to log in to their CCAvenue M.A.R.S account and get the API credentials for using these environments.

All transactions initiated by the merchant on our test environment are not processed. Test environment is strictly for testing the request and response functions.

After successfully testing the integration, merchant can move to the production environment by changing the URL.

CCAvenue test URL is: <https://test.ccavenue.com>

CCAvenue production URL is: <https://secure.ccavenue.com>

To test the integration login to your CCAvenue M.A.R.S account, under Settings tab -> API Keys page; copy the following credentials:

1. Merchant ID
2. Access Code
3. Working Key

Integration Methods

CCAvenue supports collecting payment information using following methods. All methods are designed to support a seamless user-experience.

1. **CCAvenue billing page (Non-Seamless)** - Avoid the hassle of developing and managing your own checkout page. Use the customizable billing page provided by CCAvenue which enables you to collect billing and shipping information of the customer.
2. **CCAvenue iFrame Checkout** - Fastest and easiest way to enable payments on your website. CCAvenue iframe checkout is a pre-configured form, which validates the payment data, allows user to store card information to expedite the payment process in future. CCAvenue iFrame checkout also handles PCI compliance.
3. **Custom checkout form** - Merchants can build a custom checkout form to collect order and payment information and pass the same to CCAvenue server for payment processing. CCAvenue can also store the payment information of the customer to expedite the payment process in future.
4. **CCAvenue shopping cart** - CCAvenue provides merchants with a product management module and a customizable shopping cart thereby eliminating the need for developing/maintaining their own.

Processing orders using CCAvenue billing page

CCAvenue billing page helps you avoid the hassle of developing and managing your own billing page. CCAvenue billing page is fully customizable enabling you to match the look and feel of your website.

Process flow

1. Customer selects product/service on your website and proceeds to make payment.
2. Customer is redirected to the CCAvenue billing page where billing, shipping and payment information is entered by the customer.
3. On submission of the transaction information, CCAvenue initiates the authorization process by connecting to the relevant bank/processing organization.
4. On receiving the authorization status from the bank, CCAvenue sends the response back to your website with the transaction status.

Basic steps involved in integration with the CCAvenue billing page:

Set Up: Download the CCAvenue client library as instructed here. You will have to use the CCAvenue transaction file (e.g. ccavRequestHandler.php) to initiate the payment process.

Configure: Every merchant receives a unique set of keys for transaction processing. These need to be configured in the transaction file used to initiate the payment process.

From your MARS account under Settings tab -> API Keys page; copy the merchant id, access code and secret encryption. Set these values in the file (e.g. ccavRequestHandler.php) downloaded with the integration kit.

Payment Processing: You will have to post the order information to the CCAvenue transaction file (e.g. ccavRequestHandler.jsp) to initiate the payment process. CCAvenue transaction file on receiving the order related data will encrypt the data and forward the encrypted request to the CCAvenue billing page.

JSP

```
<html>
<head><title>Sample Transaction File</title></head>
<body>
<%@ page import = "java.io.*, com.ccavenue.transaction.util.AesCryptUtil" %>
<%@include file="libFunctions.jsp"%>
<%
    String merchant_id = "2193"; //Put your merchant id here
    String access_code = " F94007DF1640D69A"; //Put access code here
    String enc_key = "FABE114254BDBC7823534894FFFCCC1"; //Put encryption key here
    Enumeration enumeration=request.getParameterNames ();
    String ccaRequest="", pname="", pvalue="";
    while (enumeration.hasMoreElements ()) {
        pname = ""+enumeration.nextElement ();
        pvalue = request.getParameter (pname);
        ccaRequest = ccaRequest + pname + "=" + pvalue + "&";
    }
    AesCryptUtil aesUtil=new AesCryptUtil (enc_key);
    String encRequest=aesUtil.encrypt (ccaRequest);
%>
<form method="post" name="redirect" action="https://test.ccavenue.com/transaction/transaction.do?
command= initiateTransaction"/>
<input type="hidden" id="encRequest" name="encRequest" value="<%= encRequest %>">
<input type="hidden" name="access_code" id="access_code" value="<%= access_code %>">
<script language='javascript'>document.redirect.submit ();</script>
</form>
</body>
</html>
```

Request Parameters

Merchant must send the following parameters to the CCAvenue PG for processing an order.

Required Parameters		
Name	Description	Type (length)
merchant_id	Merchant Id is a unique identifier generated by CCAvenue for each activated merchant.	Numeric
order_id	This ID is used by merchants to identify the order. Ensure that you send a unique id with each request. CCAvenue will not check the uniqueness of this order id as it generates a unique payment reference number for each order which is sent by the merchant.	Alphanumeric (30) Characters allowed: Alphabet (A-Z), (a-z), Numbers,- (hyphen), / (slash), _ (underscore)
currency	Currency in which you want to process the transaction. INR – Indian Rupee USD – United States Dollar SGD – Singapore Dollar GBP – Pound Sterling EUR – Euro, official currency of Eurozone	Alphabets (3)
amount	Order amount	Numeric (12, 2)
redirect_url	CCAvenue will post the status of the order along with the parameters to this URL. If you do not send this value, order status will be sent back to the URL configured in dynamic event notifications module in your MARS account. If there is no URL configured in the MARS account, PG will display the status of the order on the CCAvenue confirmation page.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z), Numbers, / (slash), _ (underscore)
cancel_url	CCAvenue will redirect the customer to this URL if the customer cancels the transaction on the billing page.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z), Numbers, / (slash), _ (underscore)
language	CCAvenue billing page is multi-lingual. Currently we are displaying the page in English (Code - EN).	Alphabet(5)

Merchant can send any of the following parameters in addition to the required parameters.

Billing and Shipping Information		
Name	Description	Type (length)
billing_name	Name of the customer	Alphabets (60) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_address	Customer's billing address	Alphanumeric (150) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, / (slash), dot, - (hyphen) Space in between words.
billing_city	Customer's billing city	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_state	Customer's billing state	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_zip	Customer's billing zip code	Alphanumeric (15) Characters allowed: Alphabet (A-Z), (a-z). Numbers
billing_country	Customer's billing country	Alphabets (50) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_tel	Customer's phone number	Numeric (20)

billing_email	Customer's email address	Alphanumeric (70) Characters allowed: Alphabet (A-Z), (a-z). Numbers @ (at), dot, _ (underscore)
delivery_name	Recipient's name	Alphabets (60) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_address	Shipping address	Alphanumeric (150) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, / (slash), dot, - (hyphen) Space in between words.
delivery_city	Shipping city	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_state	Shipping state	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_zip	Shipping zip code	Alphanumeric (15) Characters allowed: Alphabet (A-Z), (a-z). Numbers
delivery_country	Shipping country	Alphabets (50) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_tel	Shipping phone number	Numeric (20)

merchant_param1	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param2	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param3	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param4	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param5	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
promo_code	This parameter is used for sending the code of the promotion you have created in the CCAvenue MARs by which you may offer specific discounts to customers using specific payment options.	Alphanumeric (20) Characters allowed: Alphabet (A-Z), (a-z). Numbers
tid	This parameter is used for sending the unique identifier to identify uniqueness of	Numeric(17)

	the order. This is an optional parameter. Value for this parameter can be generated using the piece of code given in the integration kit.	Characters allowed: Only numbers
--	---	--

Processing orders using CCAvenue iFrame Checkout

This is the fastest and the easiest way to enable payments on merchant website. CCAvenue iframe checkout will enable the merchants to display the payment options on their checkout page and there by collect the payment information on their checkout page.

Process flow

1. Customer after selecting the product/service and entering the shipping details will proceed to make the payment on your billing page.
2. On your billing page customer selects the payment option and enters the payment information in the CCAvenue iFrame which is loaded after submitting the order information like merchant id, amount, currency, shipping information (optional) and billing information (optional).
3. On submission of the payment information, CCAvenue initiates the authorization process by connecting to the relevant bank/processing organization.
4. On receiving the authorization status from the bank, CCAvenue sends the response back to your website with the transaction status.

Basic steps involved in integration CCAvenue iFrame into Checkout page:

Set Up: Download the CCAvenue client library as instructed here. You will have to use the CCAvenue iframe code (e.g. ccavRequestHandler.jsp for JSP) to initiate the payment process.

Configure: Every merchant receives a unique set of keys for transaction processing. These need to be configured in the transaction file used to initiate the payment process.

From your MARS account under Settings tab -> API Keys page; copy the merchant id, access code and secret encryption.

You must integrate the iFrame code into your payment page to load the iFrame. Refer to the code below. You also need to set the keys in your payment file for processing the transaction.

Payment Processing: On your payment page send a request to the CCAvenue PG with order information like merchant id, amount, currency, shipping information (optional) and billing information (optional) to load the CCAvenue iFrame. CCAvenue iFrame will display the payment options for customer to select.

For initiating the iframe add the following bit of code to your website checkout page. This code is available in the kit you download for integration purpose.

Sample JSP Code

```
<div id="paymentDiv"></div>
<iframe width="482" height="500" scrolling="No" frameborder="0" id="paymentFrame" src=""></iframe>

<script type="text/javascript">
    $(document).ready(function() {
        $(function() {
            $("#checkout").live('click', function(e) {
                $("#paymentDiv").children().remove();
                var formData = $("#form[name='customerData']").serialize();
                $.post("/transaction/jsp/iframe/iframeEncReq.jsp?", formData, function(data) {
                    var encRequest, Merchant_Id, url;
                    $("#paymentDiv").append(data);
                    Merchant_Id = $("#paymentDiv").find("#merchantId").val();
                    encRequest = $("#paymentDiv").find("#encRequest").val();
                    url = "https://secure.ccavenue.com/transaction/transaction.do?command=initiateTransaction&Merchant_Id="
                        + Merchant_Id + "&encRequest=" + encRequest;
                    $("#paymentFrame").attr("src", url);
                });
                e.preventDefault();
            });
        });

        $('iframe#paymentFrame').load(function() {
            window.addEventListener('message', function(e) {
                $("#paymentFrame").css("height", e.data['newHeight'] + 'px');
            }, false);
        });

    });
</script>
```

Request Parameters

Merchant must send the following parameters to the CCAvenue PG for initiating the transaction and loading the CCAvenue iFrame.

Required Parameters		
Name	Description	Type (length)
merchant_id	Merchant Id is a unique identifier generated by CCAvenue for each activated merchant.	Numeric
order_id	This ID is used by merchants to identify the order. Ensure that you send a unique id with each request. CCAvenue will not check the uniqueness of this order id as it generates a unique payment reference number for each order which is sent by the merchant.	Alphanumeric (30) Characters allowed: Alphabet (A-Z), (a-z), Numbers, - (hyphen), / (slash), , _ (underscore)
currency	Currency in which you want to process the transaction. INR – Indian Rupee USD – United States Dollar SGD – Singapore Dollar GBP – Pound Sterling EUR – Euro, official currency of Eurozone	Alphabets (3)
amount	Order amount	Numeric (12, 2)
redirect_url	CCAvenue will post the status of the order along with the parameters to this URL. If you do not send this value, order status will be sent back to the URL configured in dynamic event notifications module in your MARS account. If there is no URL configured in the MARS account, PG will display the status of the order on the CCAvenue confirmation page.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z), Numbers, / (slash), _ (underscore)
cancel_url	CCAvenue will redirect the customer to this URL if the customer cancels the transaction on the billing page.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z), Numbers, / (slash), _ (underscore)
integration_type	Describes the type of iFrame flow iframe_normal – the bank pages will be displayed in the same tab as the payments page.	Exact value expected iframe_normal
language	CCAvenue billing page is multi-lingual. Currently we are displaying the page in English (Code - EN).	Alphabet(5)

Merchant can send any of the following parameters in addition to the required parameters.

Billing and Shipping Information		
Name	Description	Type (length)
billing_name	Name of the customer	Alphabets (60) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_address	Customer's billing address	Alphanumeric (150) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen) Space in between words.
billing_city	Customer's billing city	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_state	Customer's billing state	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_zip	Customer's billing zip code	Alphanumeric (15) Characters allowed: Alphabet (A-Z), (a-z). Numbers
billing_country	Customer's billing country	Alphabets (50) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_tel	Customer's phone number	Numeric (20)
billing_email	Customer's email address	Alphanumeric (70) Characters allowed: Alphabet (A-Z), (a-z). Numbers @ (at), dot, _ (underscore)
delivery_name	Recipient's name	Alphabets (60)

Delivery_address	Shipping address	Alphanumeric (150) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen) Space in between words.
Delivery_city	Shipping city	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_state	Shipping state	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_zip	Shipping zip code	Alphanumeric (15) Characters allowed: Alphabet (A-Z), (a-z). Numbers
delivery_country	Shipping country	Alphabets (50) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_tel	Shipping phone number	Numeric (22) Characters allowed: Numbers and - (Hyphen)
merchant_param1	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param2	This parameter can be used for sending additional	Alphanumeric (100)

	information about the transaction. PG will send this parameter in the reconciliation report.	Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param3	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param4	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param5	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
promo_code	This parameter is used for sending the code of the promotion you have created in the CCAvenue MARs by which you may offer specific discounts to customers using specific payment options.	Alphanumeric (20) Characters allowed: Alphabet (A-Z), (a-z). Numbers

Processing orders using custom checkout form

Merchants can build a custom checkout form to collect order and payment information and pass the same to CCAvenue directly for payment processing.

Process flow

1. Customer after selecting the product/service and entering the shipping details will proceed to make the payment on your billing page.
2. On your customized billing page customer selects the payment option from the list provided by CCAvenue as a JSON object. Customer enters the payment information and submits the form.
3. On submission of the payment information, CCAvenue initiates the authorization process by connecting to the relevant bank/processing organization.
4. On receiving the authorization status from the bank, CCAvenue sends the response back to your website with the transaction status.

Basic steps involved in fetching payment options to create your custom checkout form:

Set Up: Download the CCAvenue client library as instructed here. Refer to the sample page `dataFrom.htm` for making a call to the CCAvenue server for fetching payment options allocated to your account.

Configure: Every merchant receives a unique set of keys for transaction processing. These need to be configured in the transaction file used to initiate the payment process.

From your MARS account under Settings tab > API Keys page; copy the merchant ID, access code and secret encryption.

Payment Processing: Refer to the following code and make necessary changes in your payment page for fetching the payment options JSON object from the CCAvenue server. In the integration kit you will find this code in `dataFrom.htm` file.

JSON object will contain following information:

1. **Payment Option Type** – Will contain payment options allocated to the merchant. Options may include Credit Card, Net Banking, Debit Card, Cash Cards, EMI Payments or Mobile Payments.
2. **Card Type** – Will contain card type allocated to the merchant. Options may include Credit Card, Net Banking, Debit Card, Cash Cards or Mobile Payments.
3. **Card Name** – Will contain name of card. E.g. Visa, MasterCard, American Express or and bank name in case of Net banking.
4. **Payment Mode Status** – Will help in identifying the status of the payment mode. Options may include Active or Down.
5. **Error** – This parameter will enable you to troubleshoot any configuration related issues. It will provide error description.

You will have to post the order information to the CCAvenue transaction file (e.g. `ccavRequestHandler.jsp`) to initiate the payment process. CCAvenue transaction file on receiving the order related data will encrypt the data and forward the encrypted request to the CCAvenue server for processing.

Sample Code

```
<script type="text/javascript">
$(function(){
    var jsonData;
    var access_code=""; //Put access code here
    var amount="10.00";
    var currency="INR";

    $.ajax({
        url:'https://test.ccavenue.com/transaction/transaction.do?command=getJsonData&access_code='+
            access_code+'&currency='+currency+'&amount='+amount,
        dataType: 'jsonp',
        jsonp: false,
        jsonpCallback: 'processData',
        success: function (data) {
            jsonData = data;
        },
        error: function(xhr, textStatus, errorThrown) {
            alert('An error occurred! ' + ( errorThrown ? errorThrown :xhr.status ));
        }
    });

    $("#payOption").click(function(){
        $("#card_name").children().remove(); // remove old card names from old one
        $("#card_name").append("<option value=''>Select</option>");

        var paymentOption = ${this}.val();
        $("#card_type").val(paymentOption.replace("OPT", ""));

        $.each(jsonData, function(index,value) {
            if(value.payOpt==paymentOption){
                var payOptJSONArray = $.parseJSON(value[paymentOption]);
                $.each(payOptJSONArray, function() {
                    $("#card_name").find("option:last").after("<option class='"+this['dataAcceptedAt']+" "+
                        this['status']+"'' value='"+this['cardName']+"''>"+this['cardName']+"</option>");
                });
            }
        });
    });

    $("#card_name").click(function(){
        if(${this}.find(":selected").hasClass("DOWN")){
            alert("Selected option is currently unavailable. Select another payment option or try again later.");
        }
        if(${this}.find(":selected").hasClass("CCAvenue")){
            $("#data_accept").val("Y");
        }else{
            $("#data_accept").val("N");
        }
    });
});
</script>
```

Request Parameters

Merchant must send the following parameters to the CCAvenue PG for processing an order.

Required Parameters		
Name	Description	Type (length)
merchant_id	Merchant Id is a unique identifier generated by CCAvenue for each activated merchant.	Numeric
order_id	This ID is used by merchants to identify the order. Ensure that you send a unique id with each request. CCAvenue will not check the uniqueness of this order id as it generates a unique payment reference number for each order which is sent by the merchant.	Alphanumeric (30) Characters allowed: Alphabet (A-Z), (a-z), Numbers, - (hyphen), / (slash), _ (underscore)
currency	Currency in which you want to process the transaction. INR – Indian Rupee USD – United States Dollar SGD – Singapore Dollar GBP – Pound Sterling EUR – Euro, official currency of Eurozone	Alphabets (3)
amount	Order amount	Numeric (12, 2)
redirect_url	CCAvenue will post the status of the order along with the parameters to this URL. If you do not send this value, order status will be sent back to the URL configured in dynamic event notifications module in your MARS account. If there is no URL configured in the MARS account, PG will display the status of the order on the CCAvenue confirmation page.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z), Numbers, / (slash), _ (underscore)
cancel_url	CCAvenue will redirect the customer to this URL if the customer cancels the transaction on the billing page.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z), Numbers, / (slash), _ (underscore)
payment_option	Payment option selected by the customer OPTCRDC - Credit Card OPTDBCRD - Debit Card OPTNBK - Net Banking OPTCASHC - Cash Card OPTMOBP - Mobile Payments	Alphabets (10)

card_type	Type of card used by the customer. CRDC - Credit Card DBCRD - Debit Card NBK - Net Banking CASHC - Cash Card MOBP - Mobile Payments	Alphabets (10)
card_name	Name of the card used by the customer. This list will be provided by CCAvenue.	Alphabets(100) Characters allowed: Alphabet (A-Z), (a-z).
data_accept	Resend the parameter value received at the time of fetching the payment options. Expected values – Y or N	Alphabets(1)
card_number	Card number entered by the customer.	Numeric
expiry_month	Card expiry month	Numeric
expiry_year	Card expiry year	Numeric
cvv_number	Card CVV number	Numeric
issuing_bank	Card issuing bank name	Alphabets(100) Characters allowed: Alphabet (A-Z), (a-z).
mobile_no	Mobile no (Only in case of Mobile payments.)	Numeric

Merchant can send any of the following parameters in addition to the required parameters.

Name	Description	Type (length)
billing_name	Name of the customer	Alphabets (60) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_address	Customer's billing address	Alphanumeric (150) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, / (slash), dot, - (hyphen) Space in between words.
billing_city	Customer's billing city	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_state	Customer's billing state	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_zip	Customer's billing zip code	Alphanumeric (15) Characters allowed: Alphabet (A-Z), (a-z). Numbers
billing_country	Customer's billing country	Alphabets (50) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_tel	Customer's phone number	Numeric (20)
billing_email	Customer's email address	Alphanumeric (70) Characters allowed: Alphabet (A-Z), (a-z). Numbers @ (at), dot, _ (underscore)
delivery_name	Recipient's name	Alphabets (50) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.

delivery_address	Shipping address	Alphanumeric (150) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen) Space in between words.
delivery_city	Shipping city	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_state	Shipping state	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_zip	Shipping zip code	Alphanumeric (15) Characters allowed: Alphabet (A-Z), (a-z). Numbers
delivery_country	Shipping country	Alphabets (50) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_tel	Shipping phone number	Numeric (20)
merchant_param1	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param2	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)

merchant_param3	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param4	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param5	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)

Vault feature for storing card

CCAvenue enables the merchants to store card information of their customers for future transactions. This option is available in seamless and non-seamless implementations.

CCAvenue PG needs an additional parameter to identify your customer. You can send unique ID of the customer in your system at the time of initiating the transaction. This unique ID can be a customer ID, mobile number or an email ID. CCAvenue PG will store the card information against the customer identifier.

If there are any payment options stored against a customer identifier, CCAvenue PG will retrieve and load the same for customer to make the payment. Customer will also have an option of paying through a new card/payment option.

Vault Information			
customer_identifier	The identifier against which the card information is to be stored or retrieved Email ID Customer ID Mobile number	Alphanumeric, '@' and '.' are allowed	70

Processing orders using CCAvenue shopping cart

CCAvenue shopping cart helps you avoid the hassle of developing and managing your own shopping cart. CCAvenue shopping cart is fully customizable enabling you to match the look and feel of your website.

Process flow

1. Customer views the product/service displayed on your website.
2. He selects a product by clicking on the 'add to cart' button.
3. The Customer is redirected to the CCAvenue shopping cart page where the product added is displayed.
4. On the shopping card the customer can select variants, extras and update the quantity of the product he has added.
5. The customer may opt to continue shopping and he will be taken back to your website.
6. If the customer opts to checkout he will be taken to the CCAvenue billing page where billing, shipping and payment information is entered by the customer.
7. On submission of the transaction information, CCAvenue initiates the authorization process by connecting to the relevant bank/processing organization.
8. On receiving the authorization status from the bank, CCAvenue sends the response back to your website with the transaction status.

Basic steps involved in integration with the CCAvenue shopping cart:

Set Up: To set up your shopping cart you must first log in to your CCAvenue Mars account and add products under the 'Products' tab.

Configure: From your MARS account under Products tab -> Products page; copy the product id, for the products you wish to list on your website.

You need to add **"Buy Now"** or **"Add to Cart"** buttons alongside each product displayed on your website. The appropriate product id must be assigned in each of these links or buttons.

Payment Processing: You will have to post your merchant id, product id, currency and shopping URL to add a product to the CCAvenue shopping cart page. When a customer clicks on "Continue Shopping" button on the shopping cart he will be taken to the shopping URL you have specified in the request.

When the customer clicks on checkout he is taken to the CCAvenue billing page to complete the transaction.

You need to add a **"Buy Now"** or **"Add to Cart"** link or button alongside each of the products as explained below:

```
<ahref="https://test.ccavenue.com
/transaction/txn/shopcart/access_code,product_id,currency,shopping_url/language">Buy
Now</a>
```

```
<ahref="https://test.ccavenue.com/transaction/txn/shopcart/EUAF9DAAJWHDGFY3,546,INR,htt
p://yoursite.com/shop.htm/EN"> Add To Cart </a>
```

```
<ahref="https://test.ccavenue.com/transaction/txn/shopcart/EUAF9DAAJWHDGFY3,546,INR,htt
p://yoursite.com/shop.htm/EN"><img src='add2cart.jpg'></a>
```

Request Parameters

Required Parameters		
Name	Description	Type (length)
merchant_id	Merchant Id is a unique identifier generated by CCAvenue for each activated merchant.	Numeric

product_id	This ID is used to identify the product. This is available in your CCAvenue MARS panel.	Alphanumeric (30) Characters allowed: Alphabet (A-Z), (a-z), Numbers, - (hyphen), / (slash), , _ (underscore)
currency	Currency in which you want to process the transaction. INR – Indian Rupee USD – United States Dollar SGD – Singapore Dollar GBP – Pound Sterling EUR – Euro, official currency of Eurozone	Alphabets (3)
shopping_url	CCAvenue will post the status of the order along with the parameters to this URL. If you do not send this value, order status will be sent back to the URL configured in dynamic event notifications module in your MARS account. If there is no URL configured in the MARS account, PG will display the status of the order on the CCAvenue confirmation page.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z), Numbers, / (slash), _ (underscore)
language	CCAvenue billing page is multi-lingual. Currently we are displaying the page in English (Code - EN).	Alphabet(5)

The currency and language sent in the first request will be set for the session of the shopping cart.

Response Parameters

CCAvenue PG will return following parameters:

Name	Description	Type (length)
order_id	Unique ID sent by the merchant at the time of initiating the transaction.	Alphanumeric (30) Characters allowed: Alphabet (A-Z), (a-z), Numbers, # (hash), /(slash, - (hyphen)
tracking_id	Unique payment reference number generated by CCAvenue for each order.	Numeric (12)
bank_ref_no	Reference number generated by the bank for the transaction.	Alphanumeric
order_status	Status of the order. Success Failure Aborted Invalid	Alphabets (1)
failure_message	Reason for failure.	Alphanumeric
payment_mode	The payment mode for the transaction for eg , web , ivrs, emi, netbanking, debit card .	Alphabets
card_name	Specifies the type of credit card, debit card, netbanking etc .	Alphabets
status_code	The status code for this transaction	Numeric (3)
status_message	The status message for this transaction.	Alphanumeric (150)
currency	Currency code in which the transaction was processed. INR – Indian Rupee USD – United States Dollar SGD – Singapore Dollar GBP – Pound Sterling EUR – Euro, official currency of Eurozone	Alphabets (3)
Amount	Order amount	Numeric (12, 2)

billing_name	Name of the customer	Alphabets (60) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_address	Customer's billing address	Alphanumeric (150) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen) Space in between words.
billing_city	Customer's billing city	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_state	Customer's billing state	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_zip	Customer's billing zip code	Alphanumeric (15) Characters allowed: Alphabet (A-Z), (a-z). Numbers
billing_country	Customer's billing country	Alphabets (50) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
billing_tel	Customer's phone number	Numeric (20)

billing_email	Customer's email address	Alphanumeric (70) Characters allowed: Alphabet (A-Z), (a-z). Numbers @ (at), dot, _ (underscore)
delivery_name	Recipient's name	Alphabets (60) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_address	Shipping address	Alphanumeric (150) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen) Space in between words.
delivery_city	Shipping city	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_state	Shipping state	Alphabets (30) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_zip	Shipping zip code	Alphanumeric (15) Characters allowed: Alphabet (A-Z), (a-z). Numbers

delivery_country	Shipping country	Alphabets (50) Characters allowed: Alphabet (A-Z), (a-z). Space in between words.
delivery_tel	Shipping phone number	Numeric (22)
merchant_param1	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param2	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param3	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param4	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)
merchant_param5	This parameter can be used for sending additional information about the transaction. PG will send this parameter in the reconciliation report.	Alphanumeric (100) Characters allowed: Alphabet (A-Z), (a-z). Numbers # (hash), Comma, circular brackets, /(slash), dot, - (hyphen)

vault	This parameter can be used if merchant availing the vault option. On using vault functionality if card details are saved at CCAvenue end value returned will be Y. If card details are not saved at CCAvenue end the value returned for this parameter will be N	Character(1) Characters allowed: Y or N
offer_type	This parameter can be used for sending additional information if customer has used any discount or promotion while completing the transaction. If customer is using discount-coupon, value of this parameter would be discount. If customer is using promo-code, value of this parameter would be promotion.	Alphabets (9)
offer_code	This parameter can be used for sending additional information about the discount coupon and Promo code used while completing the transaction. If customer has used Discount the value sent would be If customer has used Promotion the value sent would be Promo code	Alphanumeric (30)
discount_value	This parameter can be used for sending additional information about the discounted amount.	Numeric (12,2)



Contact Details

For any assistance in integrating CCAvenue payment gateway kindly contact:

CCAvenue Technical Support

Tel : +91 022 67425555 **Ext:** 401- 408

Email : service@ccavenue.com