

CCAvenue Integration Document



Version 3.0

January 2019

INDEX

1. CCAvenue Integration Document
2. Testing and Production Environment
3. Integration Methods
4. CCAvenue billing page (Non-Seamless)
 - a. *Basic steps involved in integration with the CCAvenue billing page*
 - b. *Request Parameters*
 - c. *Response Parameters*
5. CCAvenue iFrame Checkout
 - a. *Basic steps involved in integration CCAvenue iFrame into Checkout page*
 - b. *Request Parameters*
6. Custom checkout integration
 - a. *Basic steps involved in fetching payment options to create custom checkout form*
 - b. *Request Parameters*
7. CCAvenue “Direct Connect” Payload integration
 - a. *Basic steps involved in fetching payment options to create custom checkout form*
 - b. *Request Parameters*
8. JS Based Integration
 - a. *Basic steps involved in fetching payment options to create custom checkout form*
 - b. *Request Parameters*
9. CCAvenue shopping cart
 - a. *Basic steps involved in Shopping Cart*
 - b. *Request Parameters*
10. DCC Setup (Dynamic Currency Conversion)
 - a. *Response parameters - DCC*
11. CCAvenue Vault
12. CCAvenue API's
13. Contact Details

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 environment transactions are processed through the same URL.

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 or same can be obtained by the CCAvenue team.

All transactions initiated by the merchant on our test environment are through test card. 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 test credentials..

CCAvenue production URL is:

<https://secure.ccavenue.ae/transaction/transaction.do?command=initiateTransaction>

To test the integration login you will need the below 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 integration** - 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. This method allows the merchant to accept card details on their page and hence requires the PCI DSS certificate.
4. **CCAvenue “Direct Connect” Payload Integration** - This integration enables you to deliver payment services directly through your website without redirecting your users to CCAvenue. This integration is fast and secure. It gives you control to not only build your own custom checkout form, but also control the payment request process with the banks. You shall receive the payment data from CCAvenue and the bank link on which data needs to be posted.
5. **JS Based Integration** – This integration enables you to accept card data on your form, encrypt the same on client side and post the data to CCAvenue, the card details are not submitted to your server and are encrypted using JS scripts.
6. **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.

1. CCAvenue billing page (Non-Seamless)

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 from the MARS panel. Click on “Resources” on the navigation bar of the Dashboard and click “Integration Kit”. You will have to use the CCAvenue transaction file (e.g. ccavRequestHandler.php) to initiate the payment process.

Alternatively you can obtain the integration kit and test credentials by contacting our service team.

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.

SAMPLE 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://secure.ccavenue.ae/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. AED – Dirhams The currency posted by you needs to be activated with CCAvenue for your account.	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. General validation rules apply for all below 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 the order. This is an optional parameter. Value for this parameter can be generated using the piece of code given in the integration kit. <i>The uniqueness of TID is valid for 24 hours only.</i>	Numeric(17) Characters allowed: Only numbers
response_type	You may post this parameter in case you need the response parameters from CCAvenue in JSON format. OPTIONAL PARAMETER	String (JSON)
saveCard	This parameter is required only if you are using Seamless integration/JS integration and posting card details from your end. This parameter states CCAvenue to save the card details against the customer_identifier for future use.	String (Y or N)
customer_identifier	You may post a value for this parameter for using the CCAvenue Vault feature and saving card details against the same in CCAvenue.	Alphanumeric – This can be the customer mobile no, email id, cust id etc.

Response Parameters

Name	Description	Type (length)
order_id	Order 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 (Auth Code) generated by the bank for the transaction.	Alphanumeric
order_status	Status of the order. Success Failure Aborted Invalid	Alphabets (1)
failure_message	You shall not receive any value in this parameter.	NA
payment_mode	The payment mode for the transaction for eg Credit Card, Debit Card.	Alphabets
card_name	Specifies the type of credit card, debit card ie MasterCard, Visa etc	Alphabets
status_code	The status code posted by the bank for transaction, these are the standard banks code	Numeric (2)
status_message	The status message posted by the bank for transaction, this parameter includes the decline reason in case posted by bank.	Alphanumeric (150)
currency	Currency code in which the transaction was processed.	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 is used if merchant is 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 is 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 is 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 the discount code If customer has used Promotion the value sent would be Promo code	Alphanumeric (30)
discount_value	This parameter is used for sending additional information about the discounted amount.	Numeric (12,2)
mer_amount	The actual amount posted by merchant in real time request.	Numeric (2 decimals)
eci_value	This parameter implies the 3D secure nature of transaction. 05 : Complete 3D secure transaction. 06: Partial 3D secure transaction, either the card holder or issuing bank is not participating. 07: Complete non 3D secure transaction	Numeric (2)

card_holder_name	This parameter provides the card holder name provided by customer on payment page. This parameter is optional on payment page of CCAvenue.	String
bank_receipt_no	This is the reference no generated by bank for transaction.	Numeric (12)
merchant_param6	This parameter provides the first six and last four digit of card no used for transaction as a single value. Same is by default not available.	Numeric (10)

2. 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.
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 from the MARS panel. Click on “Resources” on the navigation bar of the Dashboard and click “Integration Kit”. You will have to use the CCAvenue transaction file (e.g. ccavRequestHandler.php) to initiate the payment process.

Alternatively you can obtain the integration kit and test credentials by contacting our service team.

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) along with billing details 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.

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. AED – Dirhams The currency posted by you needs to be activated with CCAvenue for your account.	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. General validation rules apply for all below parameters.

Billing and Shipping (Billing data needs to be posted for iframe Integration)		
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 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 the order. This is an optional parameter. Value for this parameter can be generated using the piece of code given in the integration kit. <i>The uniqueness of TID is valid for 24 hours only.</i>	Numeric(17) Characters allowed: Only numbers
response_type	You may post this parameter in case you need the response parameters from CCAvenue in JSON format. OPTIONAL PARAMETER	String (JSON)
saveCard	This parameter is required only if you are using Seamless integration/JS integration and posting card details from your end. This parameter states CCAvenue to save the card details against the customer_identifier for future use.	String (Y or N)
customer_identifier	You may post a value for this parameter for using the CCAvenue Vault feature and saving card details against the same in CCAvenue.	Alphanumeric – This can be the mobile no, email id, cust id

RESPONSE PARAMETERS – Kindly [click here](#) for response parameters for Iframe Integration

3. Custom checkout Integration

Merchants can build a custom checkout form to collect order and payment information and pass the same to CCAvenue directly for payment processing. This integration allows you to accept and process card details on your server hence requires your site to be PCI DDS compliant.

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 provides the payment information/card details and submits the form. The payment options for your account can be fetched by you from CCAvenue using a JSON call, the call returns you all the active payment options for your CCAvenue account.
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 from the MARS panel. Click on “Resources” on the navigation bar of the Dashboard and click “Integration Kit”. You will have to use the CCAvenue transaction file (e.g. ccavRequestHandler.php) to initiate the payment process

Alternatively you can obtain the integration kit and test credentials by contacting our service team.

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.

JSON object will contain following information: (Details for the JSON call are available in dataFrom.htm file)

1. **Payment Option Type** – Will contain payment options allocated to the merchant. Options may include Credit Card, Debit Card or any other payment options.
2. **Card Type** – Will contain card type allocated to the merchant. Options may include Credit Card, Debit Card.
3. **Card Name** – Will contain name of card. E.g. Visa, MasterCard, American Express.
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.

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. AED – Dirhams The currency posted by you needs to be activated with CCAvenue for your account.	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	Alphabets (10)
card_type	Type of card used by the customer. CRDC - Credit Card DBCRD - Debit Card	Alphabets (10)
card_name	Name of the card used by the customer. This list will be provided by CCAvenue. VISA – Visa Card MasterCard – MasterCard credit/Debit card	Alphabets(100) Characters allowed: Alphabet (A-Z), (a-z).
data_accept	Resend the parameter value received at the time of fetching the payment options in JSON Call. Expected values – Y or N	Alphabets(1)
card_number	Card number entered by the customer.	Numeric
expiry_month	Card expiry month	Numeric (MM)
expiry_year	Card expiry year	Numeric (YYYY)
cvv_number	Card CVV number	Numeric
issuing_bank	Card issuing bank name	Alphabets(100) Characters allowed: Alphabet (A-Z), (a-z).

Merchant can send any of the following parameters in addition to the required parameters. General validation rules apply for all below 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)

tid	This parameter is used for sending the unique identifier to identify uniqueness of the order. This is an optional parameter. Value for this parameter can be generated using the piece of code given in the integration kit. The uniqueness of TID is valid for 24 hours only.	Numeric(17) Characters allowed: Only numbers
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
response_type	You may post this parameter in case you need the response parameters from CCAvenue in JSON format. OPTIONAL PARAMETER	String (JSON)
saveCard	This parameter is required only if you are using Seamless integration/JS integration and posting card details from your end. This parameter states CCAvenue to save the card details against the customer_identifier for future use.	String (Y or N)
customer_identifier	You may post a value for this parameter for using the CCAvenue Vault feature and saving card details against the same in CCAvenue.	Alphanumeric – This can be the mobile no, email id, cust id

RESPONSE PARAMETERS – Kindly [click here](#) for response parameters.

4. CCAvenue “Direct Connect” Payload Integration

This integration will enable you to deliver payment services directly through your website without redirecting your users to CCAvenue. This integration is fast and secure. It gives you control to not only build your own custom checkout form, but also control the payment request process with the banks.

Process flow

1. Customer after selecting the product/service and entering the shipping details will proceed to make the payment using 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, merchant initiates a server-to-server call to CCAvenue to fetch the request payload for the payment option selected by the user.
4. The request payload received from CCAvenue will be used by you to connect directly to the bank's authentication/3D secure page, bypassing CCAvenue.
5. CCAvenue will receive the authentication status from the bank and in turn post the transaction status back to the merchant's website.

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

Set Up: Download the CCAvenue client library from the MARS panel. Click on “Resources” on the navigation bar of the Dashboard and click “Integration Kit”. You will have to use the CCAvenue transaction file (e.g. ccavRequestHandler.php) to initiate the payment process.

Alternatively you can obtain the integration kit and test credentials by contacting our service team.

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.

JSON object will contain following information:

1. **Payment Option Type** – Will contain payment options allocated to the merchant. Options may include Credit Card, Debit Card.
2. **Card Type** – Will contain card type allocated to the merchant. Options may include Credit Card, Debit Card.
3. **Card Name** – Will contain name of card. E.g. Visa, MasterCard, American Express.
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 for handling Payload at your end:

For HTML : Data returned in HTML payload is directly rendered on the browser thereby redirecting customer to bank's page.

```
HttpClient vClient = new HttpClient();
String vResponse =
vClient.processUrlConnectionReq("encRequest="+encRequest+"&access_code="+access_code,"https:/
/secure.ccavenue.ae/transaction/transaction.do?command=initiatePayloadTransaction");
out.print(vResponse); //writes payload on browser to open bank page
```

For JSON: Payload Data is returned in JSON format along with bank URL, you will need to submit the data to bank's URL along with the returned parameters.

```
HttpClient vClient = new HttpClient();
String vResponse =
vClient.processUrlConnectionReq("encRequest="+encRequest+"&access_code="+access_code,"https:/
/secure.ccavenue.ae/transaction/transaction.do?command=initiatePayloadTransaction"); //make
s server to server call to CCAvenue and recives payload in JSON fromat
String vHtml = "<!DOCTYPE html PUBLIC '-//W3C//DTD XHTML 1.0 Transitional//EN'
'http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd'>"
+ "<html xmlns='http://www.w3.org/1999/xhtml'>"
+ "<head>" + "<meta http-equiv='Content-Type' content='text/html;
charset=utf-8' />" + "<title>CCAvenue-Transaction page</title>" + "<link rel='SHORTCUT ICON'
type='image/ico' href='"
+ "/images/favicon.ico' />"
+ "<script language='javascript'>window.history.forward(); function
noBack() { window.history.forward(); } function SubmitMe(){
document.getElementById('submit').style.visibility='hidden';document.getElementById('submit').click()
; }</script>"
+ "</head>"
+ "<body style='margin:0px;' onLoad='noBack();SubmitMe();'>";
JSONObject obj = new JSONObject(vResponse);
vHtml=vHtml+"<form name='MalltoEpay' method='"+obj.get("method")+"'
action='"+obj.get("bankUrl")+"'>";
JSONObject requestData = obj.getJSONObject("data");
Iterator vKeys = requestData.keys();
while(vKeys.hasNext()){
String key = (String)vKeys.next();
vHtml = vHtml+"<input type='text' name='"+key+"' value='"+requestData.get(key)+"'>"; }
vHtml = vHtml+"<input type='submit' id='submit' value='Continue' style='display:none;'></form>"
+ "</body></html>";
out.print(vHtml);
```

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. AED – Dirhams The currency posted by you needs to be activated with CCAvenue for your account.	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	Alphabets (10)
card_type	Type of card used by the customer. CRDC - Credit Card DBCRD - Debit Card	Alphabets (10)
card_name	Name of the card used by the customer. This list will be provided by CCAvenue. VISA – Visa Card MasterCard – MasterCard credit/Debit card	Alphabets(100) Characters allowed: Alphabet (A-Z), (a-z).
data_accept	Resend the parameter value received at the time of fetching the payment options in JSON call. Expected values – Y or N	Alphabets(1)
card_number	Card number entered by the customer.	Numeric
expiry_month	Card expiry month	Numeric (MM)
expiry_year	Card expiry year	Numeric (YYYY)
cvv_number	Card CVV number	Numeric
issuing_bank	Card issuing bank name	Alphabets(100) Characters allowed: Alphabet (A-Z), (a-z).

Merchant can send any of the following parameters in addition to the required parameters. General validation rules apply for all below 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)

device_parameter	This optional parameter is used only in case Direct Connect integration for in which merchant sends a device type though which transaction is processed.	Alphabets (3) Characters allowed: MOB PC
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)

tid	This parameter is used for sending the unique identifier to identify uniqueness of the order. This is an optional parameter. Value for this parameter can be generated using the piece of code given in the integration kit. The uniqueness of TID is valid for 24 hours only.	Numeric(17) Characters allowed: Only numbers
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
response_type	You may post this parameter in case you need the response parameters from CCAvenue in JSON format. OPTIONAL PARAMETER	String (JSON)
saveCard	This parameter is required only if you are using Seamless integration/JS integration and posting card details from your end. This parameter states CCAvenue to save the card details against the customer_identifier for future use.	String (Y or N)
customer_identifier	You may post a value for this parameter for using the CCAvenue Vault feature and saving card details against the same in CCAvenue.	Alphanumeric – This can be the mobile no, email id, cust id

RESPONSE PARAMETERS – Kindly [click here](#) for response parameters.

5. JS Based Integration

Merchants can build a custom checkout form to collect order and payment information and pass the same to CCAvenue directly for payment processing. This integration allows you to accept and process card details on client side instead of submitting the card details on your server. The encryption of card details and other parameters are done on client side using JS libraries.

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 provides the payment information/card details and submits the form. The payment options for your account can be fetched by you from CCAvenue using a JSON call, the call returns you all the active payment options for your CCAvenue account.
3. The card details and other parameters are encrypted on client side using the fetched key and posted to CCAvenue for authorization.
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: Kindly contact the CCAvenue service team for the JS based integration kit.

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.

JSON Script to fetch the payment options live for your account.

Payment Processing: ccavRequestHandler.js File contains the script to generate the Hash Key. As encryption takes place at clients side (i.e in ccavRequestHandler.js) , for every request new key is generated from CCAvenue server. This key is valid for 1 minute only.

You need to provide the Access code in the same.

JSON object will contain following information: (Details for the JSON call are available in dataFrom.htm file)

6. **Payment Option Type** – Will contain payment options allocated to the merchant. Options may include Credit Card, Debit Card or any other payment options.
7. **Card Type** – Will contain card type allocated to the merchant. Options may include Credit Card, Debit Card.
8. **Card Name** – Will contain name of card. E.g. Visa, MasterCard, American Express.
9. **Payment Mode Status** – Will help in identifying the status of the payment mode. Options may include Active or Down.
10. **Error** – This parameter will enable you to troubleshoot any configuration related issues. It will provide error description.

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. AED – Dirhams The currency posted by you needs to be activated with CCAvenue for your account.	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	Alphabets (10)
card_type	Type of card used by the customer. CRDC - Credit Card DBCRD - Debit Card	Alphabets (10)
card_name	Name of the card used by the customer. This list will be provided by CCAvenue. VISA – Visa Card MasterCard – MasterCard credit/Debit card	Alphabets(100) Characters allowed: Alphabet (A-Z), (a-z).
data_accept	Resend the parameter value received at the time of fetching the payment options in JSON call. Expected values – Y or N	Alphabets(1)
card_number	Card number entered by the customer.	Numeric
expiry_month	Card expiry month	Numeric (MM)
expiry_year	Card expiry year	Numeric (YYYY)
cvv_number	Card CVV number	Numeric
issuing_bank	Card issuing bank name	Alphabets(100) Characters allowed: Alphabet (A-Z), (a-z).

Merchant can send any of the following parameters in addition to the required parameters. General validation rules apply for all below 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)

device_parameter	This optional parameter is used only in case Direct Connect integration for in which merchant sends a device type though which transaction is processed.	Alphabets (3) Characters allowed: MOB PC
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)

tid	This parameter is used for sending the unique identifier to identify uniqueness of the order. This is an optional parameter. Value for this parameter can be generated using the piece of code given in the integration kit. The uniqueness of TID is valid for 24 hours only.	Numeric(17) Characters allowed: Only numbers
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
response_type	You may post this parameter in case you need the response parameters from CCAvenue in JSON format. OPTIONAL PARAMETER	String (JSON)
saveCard	This parameter is required only if you are using Seamless integration/JS integration and posting card details from your end. This parameter states CCAvenue to save the card details against the customer_identifier for future use.	String (Y or N)
customer_identifier	You may post a value for this parameter for using the CCAvenue Vault feature and saving card details against the same in CCAvenue.	Alphanumeric – This can be the mobile no, email id, cust id

RESPONSE PARAMETERS – Kindly [click here](#) for response parameters.

6. 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 cart 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: Download the CCAvenue client library from the MARS panel. Click on “Resources” on the navigation bar of the Dashboard and click “Integration Kit”. You will have to use the CCAvenue transaction file (e.g. ccavRequestHandler.php) to initiate the payment process.

Alternatively you may request the integration code/test credentials by contacting service team.

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.

Shopping Cart Button Code

```
<a href=
"https://secure.ccavenue.ae/transaction/txn/shopcart/access_code,product_id,currency,shop
ping_url/language">Buy Now</a>

<a href=
"https://secure.ccavenue.ae/transaction/txn/shopcart/EUAF9DAAJWHDGFY3,546,AED,http://yo
ursite.com/shop.htm/EN"> Add To Cart </a>

<ahref=
"https://secure.ccavenue.ae/transaction/txn/shopcart/EUAF9DAAJWHDGFY3,546,AED,http://
yoursite.com/shop.htm/EN"><img src='add2cart.jpg'></a>
```

Access code: The access code is generated against your URL on which the products are placed

Product id : This ID is generated against each product once you add the same in CCAvenue backend

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. AED – Dirhams The currency posted by you needs to be activated with CCAvenue for your account.	Alphabets (3)
shopping_url	This is the 'Continue Shopping URL', once the customer click on the BUY button of your product he gets redirected to CCAvenue Shopping cart page; the page has a Continue Shopping Button using which customer may add more products from your website if he wishes so the URL should be URL to page where all your products are displayed.	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 Shopping Cart)

By default there is no real time response sent for Shopping cart transactions, the response is shown on CCAvenue page to the customer.

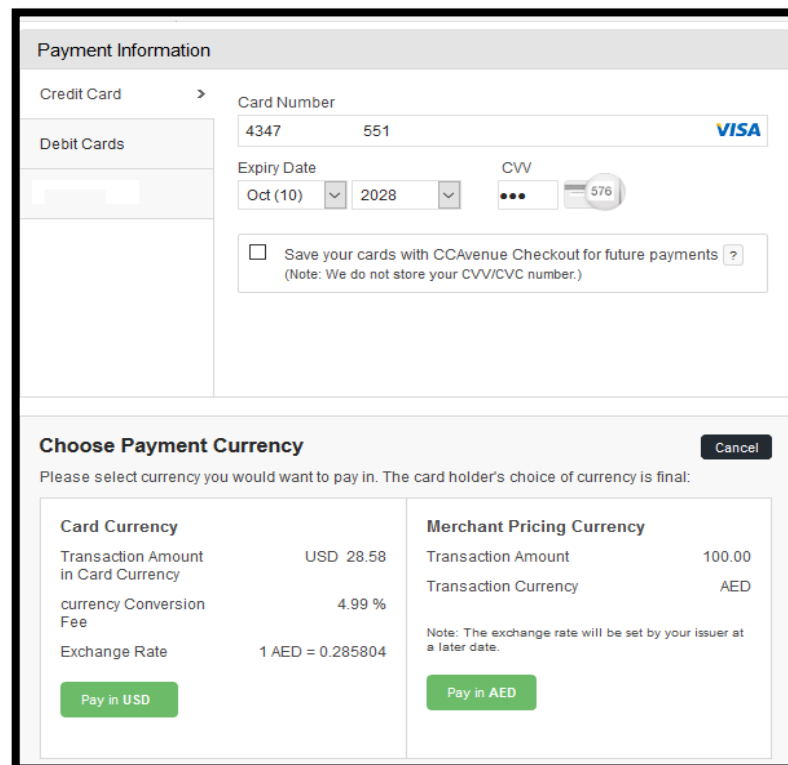
In case you need to handle the CCAvenue response in your system you need to configure the URL in CCAvenue Dynamic Event section (Kindly contact the service team for the details)

Kindly [click here](#) for response parameters.

DCC Setup (Dynamic Currency Conversion)

Activating DCC on your MID would enable your customer's to see the transaction amount in their respective card currency. DCC is currently available for bins pertaining to AED,BHD,JOD,KWD,SAR,USD currency.

- Amount is posted to CCAvenue in AED currency in your transaction request.
- On CCAvenue payment page/Iframe once the card details are provided by customer, CCAvenue shall identify the card bin country and if the card BIN pertains to one of the DCC supported countries customer shall see a DCC conversation rate below the payment section.
- Customer shall be able to see the conversation rate and can process the transaction in their card currency.



The screenshot displays the CCAvenue payment interface. The top section, titled "Payment Information", shows a credit card selection process. A "Credit Card" tab is active, displaying a Visa card with the number 4347 551. The expiry date is set to Oct (10) 2028, and the CVV is 576. A checkbox option "Save your cards with CCAvenue Checkout for future payments" is present, with a note: "(Note: We do not store your CVV/CVC number.)".

The bottom section, titled "Choose Payment Currency", prompts the user to select a currency for payment. It states: "Please select currency you would want to pay in. The card holder's choice of currency is final:". The interface is divided into two columns: "Card Currency" and "Merchant Pricing Currency".

Card Currency		Merchant Pricing Currency	
Transaction Amount in Card Currency	USD 28.58	Transaction Amount	100.00
currency Conversion Fee	4.99 %	Transaction Currency	AED
Exchange Rate	1 AED = 0.285804	Note: The exchange rate will be set by your issuer at a later date.	
Pay in USD		Pay in AED	

Additional Response parameters - DCC

dccOffered = true (Whether DCC option was selected by customer)

dccExchangeRate = 0.2858 (Exchange rate for DCC transaction)

dccForeignAmount = 28.58 (Converted amount in DCC currency)

dccForeignCurrency = USD (Currency selected by customer)

dccCurrencyConversionFee = 0499 (Conversion rate)

dccReferenceNumber = 20190109614163 (reference no by bank)

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

Server to Server API's

CCAvenue API's are designed to enable you to interact securely with our API from your client-side web application. You can get XML, JSON responses from the API, including errors.

You can use the API's to initiate a refund, capture a transaction or to update the status of any transaction along with many other actions.

All the API connections to CCAvenue (server to server) needs to be initiated from TLS 1.2 version encrypted protocol.

Kindly contact CCAvenue service team for documentation and details on same.

Contact Details

For any assistance in integrating CCAvenue payment gateway kindly contact:

CCAvenue Technical Support

Tel : +971- 45531029

Email : service@ccavenue.ae