



**Merchant's API Version 1.2**  
**Implementation Guide**

**Dated: 22<sup>nd</sup> May 2018**



## **Statement of Confidentiality**

*This document contains information that is proprietary and confidential to Avenues India Private Limited which shall not be disclosed, transmitted, or duplicated, used in whole or in part for any purpose other than its intended purpose. Any use or disclosure in whole or in part of this information without express written permission of Avenues India Private Ltd is prohibited. Any other company and product names mentioned are used for identification purposes only, and may be trademarks of their respective owners.*



## **CCAvenue API**

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

You need an active account to initiate an API call to the CCAvenue payment gateway.

### **API Authentication**

Merchant needs an active account to initiate an API call to the CCAvenue payment gateway. Merchants will have to log in to their CCAvenue M.A.R.S account and get the authentication credentials for initiating API calls.

Merchant must provide CCAvenue with the public IP address from where the API calls will be initiated. API calls will work only after CCAvenue registers the IP address provided.

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. Encryption Key



## API Calls

**CCAvenue API supports following API calls.**

1. [Status](#) – The Status API call can be used to ascertain the status of a transaction/order. You can use this call if you have not received status/information for a transaction request. It can also be used as an additional security measure to reconfirm the parameters posted back.
2. [Get Customer Payment Option](#) – The Customer payment option API call is used to list payment options saved for a customer. Payments options are saved for a customer in the vault for easy and convenient payment.
3. [Add Customer Payment Option](#) – Add Customer payment option API call is used to add another payment option for registered customer of the merchant.
4. [Payouts Summary](#) – Payouts Summary API call is used to list payouts summary for a merchant for given settlement date.
5. [PayId Details](#) - PayId Details API call is used to list transactions for a given PayId.
6. [Create Split Payout](#) – This is used to create split payout call.
7. [Split Refund](#) – This is used to refund amount that is split to sub account ID by Split Payout API.
8. [Bin Details](#) – This is used to fetch the card details for a bin number.

**Production API URL:-** <https://api.ccavenue.com/apis/servlet/DoWebTrans>

**Staging API URL:-** <https://apitest.ccavenue.com/apis/servlet/DoWebTrans>

## Encryption of Request for API Calls

Requests sent to CCAvenue will hold the parameters mentioned in the table below.  
**enc\_request**, has to be encrypted using AES similar to the method used for real-time transaction.  
 Encryption key is mapped to Access code as mentioned in API Authentication section.

Name	Description
<b>enc_request</b> ( <i>required</i> )	AES encrypted request data.
<b>access_code</b> ( <i>required</i> )	This is the access code for your application. You must send this with each request.
<b>command</b> ( <i>required</i> )	This is the command to access the API Calls. You must send this with each request.
<b>request_type</b> ( <i>required</i> )	API requests are accepted in XML, JSON or String. Specify the request type.
<b>response_type</b> ( <i>optional</i> )	API returns XML, JSON or String responses. If left blank, the response will be in the same format as request.
<b>version</b> ( <i>required</i> )	This is the version to access API based on version calls and current possible values is 1.1

### Example:

**enc\_request**=63957FB55DD6E7B968A7588763E08B240878046EF2F520C44BBC63FB9CCE726209A4734877F5904445591304ABB2F5E598B951E39EAFB9A24584B00590ADB077ADE5E8C444EAC5A250B1EA96F68D22E44EA2515401C2CD753DBA91BD0E7DFE7341BE1E7B7550&**access\_code**=8JXENNSSBEZCU8KQ&**command**=confirmOrder&**request\_type**=XML&**response\_type**=XML&**version**=1.2

## Decryption of Response for API Calls

Response received from CCAvenue will hold the parameters mentioned in the table below. enc\_response, when encrypted will have to be decrypted using AES similar to the method used for real-time transactions. Encryption key is mapped to Access code as mentioned in API Authentication section.

Name	Description
enc_response	AES encrypted response containing format as per <i>response_type</i> .
enc_error_code	enc_error_code contains value if status is "1" please refer to <a href="#">below table</a> for the error code.
status	This states whether the call was successful or not. If value of this parameter is "1" then you need not decrypt the enc_response as it will contain plain error message.

**Note:** - Please refer [below table](#) for enc\_response value when status value is "1" as follows.

### Example:

#### Successful:

**status=0&enc\_response=63957FB55DD6E7B968A7588763E08B240878046EF2F520C44BBC63FB9CCE726209A473457E6B13721EC6D05ED13A0483ACFDD6F11F284AE79755D47E79687478F93CFCD3CD97510B67B961CDB5279F209F5C451F3039696F13C990B963854C8CADF730&enc\_error\_code=**

#### Error:

**status=1&enc\_response=Access\_code: Invalid Parameter&enc\_error\_code=51407.**

## 1. Status

The Status API call can be used to ascertain the status of a transaction/order. You can use this call if you have not received status/information for a transaction request. It can also be used as an additional security measure to reconfirm the parameters posted back.

Orderstatustracker API call can be performed by following ways:

1. With both Order Number and CCAvenue Reference Number as parameters - The order which matches both number and reference number will be returned.
2. With ONLY CCAvenue Reference number as parameter - The order which matches the CCAvenue Reference number will be returned. CCAvenue Reference number is a unique number and its uniqueness is maintained by CCAvenue.
3. With ONLY Order Number as parameter - The order which matches the Order number will be returned. The uniqueness of Order Number must be maintained by the merchant and is up to his discretion. CCAvenue does not check for uniqueness of the Order Number while performing transactions. Thus, when there are multiple matching Order Numbers found, API call will return the LATEST Order.

### Request Parameters

Name	Description	Note
		Parameters Datatype (Parameters max length)
<b>enc_request</b> (required)	AES encrypted request data	
<b>access_code</b> (required)	Unique CCAvenue access code which is generated when merchant registered their IP address. You must send this with each request.	
<b>request_type</b> (required)	API requests are accepted in XML, JSON or String. Specify the request type.	Possible value for request_type is "XML" or "JSON" or "STRING".
<b>response_type</b> (optional)	API returns responses in XML, JSON or String format. If left blank, the response will be in the same format as request.	Possible value for response_type is "XML" or "JSON" or "STRING".
<b>command</b> (required)	Command value specifies the API calls. You must send this with each request.	Value is "orderStatusTracker".
<b>reference_no</b> (conditional)	CCAvenue reference no. allocated to the transaction.  Reference number is required if you do not	Numeric(25).

	share order_no.	
<b>order_no</b> ( <i>conditional</i> )	This is the merchant reference number for the transaction.  Order number is required if you do not share reference_no.	AlphaNumeric with special characters (hyphen and underscore)(30).

### Example XML Request

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Order_Status_Query order_no="33231644" reference_no="225013271813"/>
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.





### Example JSON Request

```
{  
  "reference_no": "225013271813",  
  "order_no": "33231644"  
}
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Example STRING Request

**Format:** reference\_no|order\_no|

**Example:** 225013271813|33231644|

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Response Parameters

Name	Description	Note
status	This states whether the call was successful or not. If value of this parameter is “1” then you need not decrypt the enc_response as it will contain plain error message.	Value “0” denotes that the API call was successful. Value “1” denotes API call failure. On enc_response is plain text represents the error message.
enc_response	AES encrypted response containing format as per <i>response_type</i>	
order_amt	Amount for the transaction.	Decimal(12,2).
order_bill_address	Order billing address details for the order.	Possible value for address is Alphanumeric with special characters (space, hyphen, comma, ampersand(&), hash(#), circular brackets and dot)(315).
order_bill_city	Order billing City name for the order.	Possible value for city is Alphanumeric with special characters (space, comma, hyphen and dot)(30).
order_bill_country	Order billing Country for the Order.	Possible value for country is Alphanumeric with special characters (space)(30).
order_bill_email	Email Address of the Order for notifications.	Possible value for email ID is

		Alphanumeric with special characters (hyphen, underscore, dot, @)(70).
<b>order_bill_name</b>	Order billing name for the order.	Possible value for name is Alphanumeric with special characters (space, hyphen, apostrophe, underscore, dot)(60).
<b>order_bill_state</b>	Order billing state for the order.	Alphanumeric with special characters (hyphen, dot and space)(30).
<b>order_bill_tel</b>	Order billing telephone no. for the order.	Numeric(10)
<b>order_bill_zip</b>	Order billing address's pin code for the order.	Possible value for zip is Alphanumeric with special characters (hyphen and space) (15).
<b>order_capt_amt</b>	Captured amount for the transaction. Captured amount can be full or partial of the transaction amount.	Decimal(12,2).
<b>order_curr</b>	Possible order Currency in which merchant processed the transaction.	String Examples: <b>INR</b> – Indian Rupee <b>USD</b> – United States Dollar <b>SGD</b> – Singapore Dollar <b>GBP</b> – Pound Sterling <b>EUR</b> – Euro, official currency of Eurozone
<b>order_date_time</b>	Order Generated Date &Time.	DateTime in IST( yyyy-MM-dd HH:mm:ss.SSS) format.
<b>order_device_type</b>	This is the type of device using which the transaction was processed.	Possible value for device type is <b>IVRS/MOB/PC</b> .
<b>order_discount</b>	This is Discount Value for the Order No.	Decimal(12,2).
<b>order_fee_flat</b>	Flat Fee for the Order No.	Decimal(12,2).
<b>order_fee_perc</b>	Provides the percentage fee for the same order No.	Decimal(12,2).
<b>order_fee_perc_value</b>	This attribute provides the percentage fee Value for the same order No.	Decimal(12,2).
<b>order_fraud_status</b>	Specify whether orders are valid or not.	String Possible Values are: 1) Value " <b>High</b> " denotes "High Risk" 2) Value " <b>Low</b> " denotes "Low Risk" 3) Value " <b>NR</b> " denotes "No Risk" 4) Value " <b>GA</b> " denotes "Go

		Ahead” 5) Value “ <b>NA</b> ” denotes “Not Applicable”
<b>order_gross_amt</b>	Total transaction amount.	Decimal(12,2).
<b>order_ip</b>	Customer IP Address (i.e. from where transaction is being initiated)	IP V-4 Supported.
<b>order_no</b>	Order No. for the transaction.	Alphanumeric with special characters (hyphen and underscore)(30).
<b>order_notes</b>	Order information you wish to provide.	Alphanumeric with special characters (space, comma, dot, hyphen and underscore)(60).
<b>order_ship_address</b>	Shipping Address for the order.	Possible value for address is Alphanumeric with special characters (space, hyphen, comma, ampersand(&), hash(#), circular brackets and dot)(315)
<b>order_ship_city</b>	Shipping city name for the orders.	Possible value for city is Alphanumeric with special characters (space, comma, hyphen and dot)(30).
<b>order_ship_country</b>	Shipping country name for the orders.	Possible value for country is Alphanumeric with special characters (space)(30).
<b>order_ship_email</b>	Shipping email ID for the notifications of the transaction.	Possible value for email id is Alphanumeric with special characters (hyphen, underscore, dot, @)(70).
<b>order_ship_name</b>	Shipping Name of the Customer for the order.	Possible value for name is Alphanumeric with special characters (space, hyphen, apostrophe, underscore, dot)(60).
<b>order_ship_state</b>	Shipping state for the order.	Alphanumeric with special characters (hyphen, dot and space)(30).
<b>order_ship_tel</b>	Telephone no. for notifications of the transaction.	Numeric(10).
<b>order_ship_zip</b>	Order shipping address's pin code for the order.	Possible value for zip is Alphanumeric with special characters(hyphen and space) (15).
<b>order_status</b>	Status of the order. It can be single or multiple.	String Possible values are: <b>Aborted (transaction is cancelled</b>

		<p><b>by the User)</b></p> <p><b>Auto-Cancelled</b> (transaction has not confirmed within 12 days hence auto cancelled by system)</p> <p><b>Auto-Reversed</b> (two identical transactions for same order number, both were successful at bank's end but we got response for only one of them, then next day during reconciliation we mark one of the transaction as auto reversed )</p> <p><b>Awaited</b> (transaction is processed from billing shipping page but no response is received)</p> <p><b>Cancelled</b> (transaction is cancelled by merchant )</p> <p><b>Chargeback()</b></p> <p><b>Invalid</b>(Transaction sent to CCAvenue with Invalid parameters, hence could not be processed further)</p> <p><b>Fraud</b> (we update this during recon, the amount is different at bank's end and at CCAvenue due to tampering)</p> <p><b>Initiated</b> (transaction just arrived on billing shipping page and not processed further )</p> <p><b>Refunded</b> (Transaction is refunded.)</p> <p><b>Shipped (transaction is confirmed)</b></p> <p>Successful</p> <p><b>System refund</b> (Refunded by CCAvenue for various findings of reversals by CCAvenue)</p> <p><b>Unsuccessful</b> (transaction is not successful due to )</p>
<b>order_status_date_time</b>	This is the latest date and time when order status is modified.	Date/Time in IST( yyyy-MM-dd HH:mm:ss.SSS) format.
<b>order_TDS</b>	Amount of TDS (tax deducted at source) for the Transaction.	Decimal (13,4)
<b>order_tax</b>	Tax Amount for the Transaction.	Decimal (13,4)

<b>reference_no</b>	CCAvenue reference no. allocated to the transaction.	Numeric(25).
<b>order_bank_ref_no</b>	Unique reference number shared by Bank after successful transaction.	Alphanumeric (25).
<b>order_bank_response</b>	Description about the transaction shared by the bank after transaction.	String
<b>order_gtw_id</b>	Unique payment option Bank name.	Alphabet(6)
<b>order_card_name</b>	Specify the card name for the transaction.	Possible value for card name is <b>VISA</b> ," <b>MASTERCARD</b> "," <b>AMEX</b> "," <b>JCB</b> "," <b>DINERS CLUB</b> ".
<b>order_option_type</b>	Specify the payment option type for the order.	String Possible value for payment option type is:  <b>OPTCASHC</b> -Cash card <b>OPTCRDC</b> -Credit Card <b>OPTDBCRD</b> -Debit Card <b>OPTEMI</b> -EMI <b>OPTIVRS</b> -IVRS <b>OPTMOBP</b> -MobilePayments <b>OPTNBK</b> -Net Banking
<b>param_value1</b>	Temp parameters value update by merchant at transaction time for further use.	String
<b>param_value2</b>	Temp parameters value update by merchant at transaction time for further use.	String
<b>param_value3</b>	Temp parameters value update by merchant at transaction time for further use.	String
<b>param_value4</b>	Temp parameters value update by merchant at transaction time for further use.	String
<b>param_value5</b>	Temp parameters value update by merchant at transaction time for further use.	String
<b>page_count</b>	Total pages available based on <i>no_of_records</i> in the request.	<b>Example:</b> no_of_records sent in request was 100
<b>total_records</b>	Total no. of orders matching the lookup criteria.	total_records matching the lookup criteria were 1000 page_count will be 10 (total_records / no_of_records) rounded to the ceiling.
<b>error_desc</b>	Reason if search criteria did not find the orders for the transactions.	String. Please refer <a href="#">below table</a> for failure message.
<b>error_code</b>	Error code for Failure reason.	String. Please refer <a href="#">below table</a> for

		failure message.
--	--	------------------

## Example XML Response

### Success Response:

```
<?xml version='1.0' encoding='UTF-8'?>
<Order_Status_Result error_code="">
  <error_desc></error_desc>
  <order_TDS>0.0</order_TDS>
  <order_amt>1.0</order_amt>
  <order_bank_ref_no>035944</order_bank_ref_no>
  <order_bank_response>Transaction Successful</order_bank_response>
  <order_bill_address>Room no 1101, near Railway station      Ambad</order_bill_address>
  <order_bill_city>Indore</order_bill_city>
  <order_bill_country>India</order_bill_country>
  <order_bill_email>chandrakant.patil@avenues.info</order_bill_email>
  <order_bill_name>Shashi</order_bill_name>
  <order_bill_state>MP</order_bill_state>
  <order_bill_tel>9595226054</order_bill_tel>
  <order_bill_zip>425001</order_bill_zip>
  <order_capt_amt>0.0</order_capt_amt>
  <order_card_name>MasterCard</order_card_name>
  <order_currncy>INR</order_currncy>
  <order_date_time>2015-09-16 15:05:55.573</order_date_time>
  <order_delivery_details></order_delivery_details>
  <order_device_type>PC</order_device_type>
  <order_discount>0.0</order_discount>
  <order_fee_flat>0.0</order_fee_flat>
  <order_fee_perc>2.3</order_fee_perc>
  <order_fee_perc_value>0.02</order_fee_perc_value>
  <order_fraud_status>NA</order_fraud_status>
  <order_gross_amt>1.0</order_gross_amt>
  <order_gtw_id>ICICI</order_gtw_id>
  <order_ip>192.168.2.182</order_ip>
  <order_no>66068092</order_no>
  <order_notes>order will be shipped</order_notes>
  <order_option_type>OPTCRDC</order_option_type>
  <order_ship_address>Room no 1101, near Railway station      Ambad</order_ship_address>
  <order_ship_city>Indore</order_ship_city>
  <order_ship_country>India</order_ship_country>
  <order_ship_email></order_ship_email>
  <order_ship_name>Shashi</order_ship_name>
  <order_ship_state>MP</order_ship_state>
  <order_ship_tel>9595226054</order_ship_tel>
  <order_ship_zip>425001</order_ship_zip>
```

```
<order_status>Successful</order_status>
<order_status_date_time>2015-09-16 15:06:13.243</order_status_date_time>
<order_tax>0.0028</order_tax>
<param_value1>Mobile No9595226054</param_value1>
<param_value2>Flight from Dehli</param_value2>
<param_value3>ToMumbai</param_value3>
<param_value4>Mobile No9595226054</param_value4>
<param_value5>Mobile No9595226054</param_value5>
<reference_no>204000163469</reference_no>
<status>0</status>
</Order_Status_Result>
```

### Failure Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Order_Status_Result error_code="51313">
  <error_desc>Order List: Invalid Parameter</error_desc>
  <status>1</status>
</Order_Status_Result>
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

### Example JSON Response

#### Success Response:

```
{
  "reference_no":"204000163469",
  "order_no":"66068092",
  "order_currncy":"INR",
  "order_amt":1.0,
  "order_date_time":"2015-09-16 15:05:55.573",
  "order_bill_name":"Shashi",
  "order_bill_address":"Room no 1101, near Railway station Ambad",
  "order_bill_zip":"425001",
  "order_bill_tel":"9595226054",
  "order_bill_email":"chandrakant.patil@avenues.info",
  "order_bill_country":"India",
  "order_ship_name":"Shashi",
  "order_ship_address":"Room no 1101, near Railway station Ambad",
  "order_ship_country":"India",
  "order_ship_tel":"9595226054",
```



```

"order_bill_city":"Indore",
"order_bill_state":"MP",
"order_ship_city":"Indore",
"order_ship_state":"MP",
"order_ship_zip":"425001",
"order_ship_email":"",
"order_notes":"order will be shipped",
"order_ip":"192.168.2.182",
"order_status":"Successful",
"order_fraud_status":"NA",
"order_status_date_time":"2015-09-16 15:06:13.243",
"order_capt_amt":0.0,
"order_card_name":"MasterCard",
"order_delivery_details":"",
"order_fee_perc":2.3,
"order_fee_perc_value":0.02,
"order_fee_flat":0.0,
"order_gross_amt":1.0,
"order_discount":0.0,
"order_tax":0.0028,
"order_bank_ref_no":"035944",
"order_gtw_id":"ICICI",
"order_bank_response":"Transaction Successful",
"order_option_type":"OPTCRDC",
"order_TDS":0.0,
"order_device_type":"PC",
"param_value1":"Mobile No9595226054",
"param_value2":"Flight from Dehli",
"param_value3":"ToMumbai",
"param_value4":"Mobile No9595226054",
"param_value5":"Mobile No9595226054",
"error_desc":"",
"status":0,
"error_code":""
}

```

#### Failure Response:

```

{
  "error_desc":"Order List: Invalid Parameter",
  "error_code":"51313",
  "status":1
}

```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to



the decryption section.

## Example STRING Response

### Format:

status|order\_status|reference\_no|order\_bank\_ref\_no|order\_bank\_response|  
 order\_bill\_name|order\_bill\_email|order\_bill\_address|order\_bill\_city|order\_bill\_state|order\_bill\_co  
 untry|order\_bill\_telephone\_no|order\_bill\_city\_zip|order\_card\_name|order\_currency|order\_date\_  
 time|order\_delivery\_details|order\_device\_type|order\_fraud\_status|order\_gateway\_id|order\_ip|or  
 der\_no|  
 order\_notes|order\_option\_type|order\_shipping\_name|order\_ship\_email|order\_ship\_address|order  
 \_ship\_city|order\_ship\_state|order\_ship\_country|order\_ship\_telephone\_no|order\_ship\_zip|order\_  
 status\_date\_time|order\_TDS|order\_amount|order\_capture\_amount|order\_discount|order\_fee\_fla  
 t|order\_fee\_perc|order\_fee\_perc\_value|order\_gross\_amount|order\_tax|param\_value1|param\_val  
 ue2|param\_value3|param\_value4|param\_value5|

### Example:

0|Successful|204000163514|068406|Transaction Successful|Shashi|gzpmgexii@i.softbank.jp|Room  
 no 1101, near Railway station  
 Ambad|Indore|MP|India|9595226054|425001|MasterCard|INR|2015-09-18  
 12:53:40.407||PC|NA|ICICI|192.168.2.182|64807533|order will be  
 shipped|OPTCRDC|Shashi||Room no 1101, near Railway station  
 Ambad|Indore|MP|India|9595226054|425001|2015-09-18  
 12:54:15.357|0.0|1.0|0.0|0.0|0.0|2.3|0.02|1.0|0.0028|Mobile No9595226054|Flight from  
 Dehli|ToMumbai|Mobile No9595226054|Mobile No9595226054|

### Failure Response:

**Format:** statud|error\_code|error\_desc|

**Example:** 1|51313|Order List: Invalid Parameter|

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

## 2. Get Customer Payment Options

The Customer payment option API call is used to list payment option saved for a customer. Payments options are saved for a customer in vault for easy and convenient payment.

### Request Parameters

Name	Description	Note
<b>enc_request</b> (required)	AES encrypted request data.	
<b>access_code</b> (required)	Unique CCAvenue access code which is generated when merchant registered their IP address. You must send this with each request.	
<b>request_type</b> (required)	API requests are accepted in XML, JSON or String. Please specify the request type.	Possible value for request_type is "XML" or "JSON" or "STRING".
<b>response_type</b> (optional)	API returns responses in XML, JSON or String format. If left blank, the response will be in the same format as request.	Possible value for response_type is "XML" or "JSON" or "STRING".
<b>Command</b> (required)	Command value specifies the API calls. You must send this with each request.	Possible value for this API call is "getCustomerPaymentOptions"
<b>customer_id</b> (required)	Unique Customer ID for the transaction.	Numeric(25).

### Example XML Request

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <Customer_Payment_Options
    customer_id="123"/>
```

**Note:** You will have to encrypt the above request and store in the "enc\_request" parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Example JSON Request

```
{
  "customer_id": "1234"
}
```

**Note:** You will have to encrypt the above request and store in the "enc\_request" parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Example String Request

**Format:** customer\_id|

**Example:** 6|

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

## Response Parameters

Name	Description	Note
<b>status</b>	This states whether the call was successful or not. If value of this parameter is “1” then you need not decrypt the enc_response as it will contain plain error message.	Value “0” denotes that the API call was successful. Value “1” denotes API call failure. On enc_response is plain text represents the error message.
<b>enc_response</b>	AES encrypted response containing format as per <i>response_type</i> .	
<b>customer_id</b>	Unique Customer ID for the transaction.	Numeric(25).
<b>customer_card_no</b>	Last four digit of card no for the transaction.	Numeric(4)  Will contain values only for Credit Card and Debit Card.
<b>customer_payopt_type</b>	Number of Payment options assign against to this customer id.	String.  Values : <b>OPTNBK</b> -net banking <b>OPTCRD</b> -credit card <b>OPTDBCRD</b> -debit card <b>OPTMOBP</b> -mobile payment <b>OPTIVRS</b> - IVRS <b>OPTWLT</b> - Wallet <b>OPTCASHC</b> -Cash Card <b>OPTEMI</b> - EMI
<b>customer_card_name</b>	Customer card name for the transaction.	Alphanumeric with special characters (space, hyphen, apostrophe, underscore, dot)(60).
<b>customer_email</b>	Customer email id for the transaction.	Alphanumeric with special characters (hyphen, underscore, dot,@)(70).
<b>customer_card_label</b>	Customer card label name for the transaction.	Alphanumeric with special characters (space, hyphen, apostrophe, underscore, dot)(60).
<b>customer_card_id</b>	Unique Customer Card ID is the identifier for the payment options against the Customer ID.	Numeric(25).

<b>customer_phone_number</b>	Unique Customer phone number for the transaction.	Numeric(10).
<b>customer_card_expiry</b>	Customer card expiry month and year.	Numeric with special characters.
<b>customer_card_issuing_bank</b>	Customer card issuing bank name.	String.
<b>error_desc</b>	Reason if API call does not find the record based on given search criteria.	String. Please refer <a href="#">below table</a> for failure message.
<b>error_code</b>	Error code for Failure reason.	String. Please refer <a href="#">below table</a> for failure message.

### Example XML Response

#### Success:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Customer_Payment_Option_Result customer_id="1234" error_desc="" error_code="">
  <pay_Opt_List>
    <customer>
      <customer_card_id>26</customer_card_id>
      <customer_card_no>4567</customer_card_id>
      <customer_card_name>Development Credit Bank</customer_card_name>
      <customer_card_type>NBK</customer_card_type>
      <customer_email>xxxx@xxx.com</customer_email>
      <customer_payopt_type>OPTNBK</customer_payment_type>
      <customer_card_label>xxxxxxx</customer_card_label>
      <customer_card_expiry>10/2020</customer_card_expiry>
      <customer_card_issuing_bank>xxxxxxx</customer_card_issuing_bank>
    </customer>
    <customer>
      <customer_card_id>27</customer_card_id>
      <customer_card_no>1234</customer_card_no>
      <customer_card_name>Development Credit Bank</customer_card_name>
      <customer_card_type>NBK</customer_card_type>
      <customer_email>xxxx@xxx.com</customer_email>
      <customer_payopt_type>OPTNBK</customer_payopt_type>
      <customer_card_label>xxxxxxx</customer_card_label>
      <customer_card_expiry>10/2020</customer_card_expiry>
      <customer_card_issuing_bank>xxxxxxx</customer_card_issuing_bank>
    </customer>
  </pay_Opt_List>
</Customer_Payment_Option_Result>
```

### Failure:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Customer_Payment_Option_Result
    error_desc="Customer id: Invalid Parameter" customer_id="1234" error_code="51327"/>
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

### Example JSON Response

#### Success:

```
{
  "pay_Opt_List": [{
    "customer_card_name": "Development Credit Bank",
    "customer_card_type": "NBK",
    "customer_card_id": 26,
    "customer_email": "xxxx@xxx.com",
    "customer_payopt_type": "OPTNBK",
    "customer_card_label": "xxxxx",
    "customer_card_no": "1234",
    "customer_card_expiry": "10/2020",
    "customer_card_issuing_bank": "xxxxxxx",
  },
  {
    "customer_card_name": "MasterCard",
    "customer_card_type": "CRDC",
    "customer_card_id": 27,
    "customer_email": "xxxx@xxx.com",
    "customer_payopt_type": "OPTCRDC",
    "customer_card_label": "xxxxx",
    "customer_card_no": "4567",
    "customer_card_expiry": "10/2020",
    "customer_card_issuing_bank": "xxxxxxx",
  }],
  "customer_id": "1234",
  "error_desc": "",
  "error_code": ""
}
```

#### Failure:

```
{
```



```
"error_desc": "Customer id: Invalid Parameter",  
"error_code": "51327",  
"customer_id": "1234"  
}
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

### Example STRING Response

#### Success:

**Format:** customer\_card\_id\$customer\_card\_no(last four digits)\$customer\_card\_name\$customer\_card\_type\$customer\_payopt\_type\$customer\_card\_label\$customer\_email\$customer\_card\_expiry\$customer\_card\_issuing\_bank^customer\_card\_id\$customer\_card\_no(last four digits)\$customer\_card\_name\$customer\_card\_type\$customer\_payopt\_type\$customer\_card\_label\$customer\_email\$customer\_card\_expiry\$customer\_card\_issuing\_bank|

**Example:**27\$”1234”\$MasterCard\$CRDC\$OPTCRDC\$”xxxx”\$”xxxx@xxx.com”^29\$”4567”\$MasterCard\$CRDC\$ OPTCRDC\$”xxxx”\$”xxxx@xxx.com”\$”10/2020”\$”xxxx”

#### Failure:

**Format:** error\_code|error\_desc|

**Example:** 51327|Customer id: Invalid Parameter|

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.



### 3. Add Customer Payment Option

The Add Customer Payment Option API call is used to add another payment option for the customer of the merchant.

#### Request Parameters

Name	Description	Note
<b>enc_request</b> (required)	AES encrypted request data	
<b>access_code</b> (required)	Unique CCAvenue access code which is generated when merchant registered their IP address. You must send this with each request.	
<b>request_type</b> (required)	API requests are accepted in XML, JSON or String. Specify the request type.	Possible value for request_type is "XML" or "JSON" or "STRING".
<b>response_type</b> (optional)	API returns responses in XML, JSON or String format. If left blank, the response will be in the same format as request.	Possible value for response_type is "XML" or "JSON" or "STRING".
<b>Command</b> (required)	Command value specifies the API calls. You must send this with each request.	Possible value for this API call is "addCustomerPaymentOption".
<b>customer_id</b> (required)	Unique Customer ID for the transaction.	Numeric(25).
<b>customer_name_on_card</b> (optional)	Customer name on card.	Alphanumeric with special characters (space, underscore)(30).
<b>customer_phone_no</b> (optional)	Customer mobile number.	Numeric(10).
<b>customer_email</b> (optional)	Customer email id.	Alphanumeric with special characters (hyphen, underscore, dot,@)(70).
<b>customer_card_issuer_bank</b> (optional)	Customer card issuer bank name.	Alphanumeric with special characters (space, underscore)(70).
<b>customer_card_expiry</b> (required)	Customer Card expiry date.	Date format in <b>MM/yyyy</b> .
<b>customer_card_no</b>	Customer Card number.	Numeric(25)

(required)		
<b>customer_card_name</b> (required)	Customer Card name.	String VISA MASTERCARD AMEX JCB ECRD DINERS CLUB DSNV CTBL CVMS AMEX EZE CLICK
<b>customer_card_type</b> (required)	Customer card type to payment option for the transaction.	String. CRDC-credit card DBRD-debit card

### Example XML Request

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <Add_Customer_Payment_Option_Query>
    <customer_id>14</customer_id>
    <customer_email>subash.yadav@avenues.info</customer_email>
    <customer_phone_no>8698319931</customer_phone_no>
    <customer_name_on_card>subash yadav</customer_name_on_card>
    <customer_card_name>mastercard</customer_card_name>
    <customer_card_no>421578965236545</customer_card_no>
    <customer_card_expiry>11/2015</customer_card_expiry>
    <customer_card_issuer_bank>Kotakmahindra</customer_card_issuer_bank>
    <customer_card_type>CRDC</customer_card_type>
  </Add_Customer_Payment_Option_Query>
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

## Example JSON Request

```
{
  "customer_id":14,
  "customer_email":"subash.yadav@avenues.info",
  "customer_phone_no":8698319931,
  "customer_name_on_card":"subash yadav",
  "customer_card_no":421578965236545,
  "customer_card_name":"mastercard",
  "customer_card_expiry":"11/2015",
  "customer_card_issuer_bank":"Kotak mahindra",
  "customer_card_type":"CRDC"
}
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

## Example STRING Request

### Format:

customer\_id|customer\_email|customer\_phone\_no|customer\_name\_on\_card|customer\_card\_no|customer\_card\_name|customer\_card\_expiry|customer\_issuer\_bank|customer\_card\_type|

**Example:** [14|subash.yadav@avenues.info](mailto:subash.yadav@avenues.info)|8698319931|subash yadav|421578965236545|mastercard|11/2015|Kotak mahindra|CRDC|

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

## Response Parameters

Name	Description	Note
status	This states whether the call was successful or not. If value of this parameter is “1” then you need not decrypt the enc_response as it will contain plain error message.	Value “0” denotes that the API call was successful. Value “1” denotes API call failure. On enc_response is plain text representing the error message.
enc_response	AES encrypted response containing format as per <i>response_type</i>	
customer_id	Unique Customer ID for the transaction.	Numeric(25).
customer_card_id	Unique Customer card ID against the payment option for the transaction.	Numeric (25).
Status	Delete Status value specifies whether	Possible values for this is

	customer payment option is going to be deleted successfully or not.	0 - Deletion successful. 1 - Could not be deleted.
<b>customer_card_no_last 4digits</b>	Last four digit of card no for the transaction.	Numeric(4)  Will contain values only for Credit Card and Debit card.
<b>customer_pay_opt_type</b>	Customer payment option for given details.	String  <b>OPTCRD</b> -credit card <b>OPTDBCRD</b> -debit card
<b>customer_card_expyr</b>	Customer card expiry date.	Date format in <b>MM/yyyy</b> .
<b>customer_card_issuer_bank</b>	Customer Card issuer bank.	Alphanumeric with special characters (space, underscore )(70).
<b>customer_card_name</b>	Customer Card name.	String <b>VISA</b> <b>MASTERCARD</b> <b>AMEX</b> <b>JCB</b> <b>ECRD</b> <b>DINERS CLUB</b> <b>DSNV</b> <b>CTBL</b> <b>CVMS</b> <b>AMEX EZE CLICK</b>
<b>error_desc</b>	Reason if customer payment option is not going to be deleted successfully.	String. Please refer <a href="#">below table</a> for failure message.
<b>error_code</b>	Error code for Failure reason.	String. Please refer <a href="#">below table</a> for failure message.

### Example XML Response

#### Success:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Add_Customer_Payment_Option_Result>
  <customer_card_expyr>11/2015</customer_card_expyr>
  <customer_card_id>240</customer_card_id>
  <customer_card_issuer_bank>Kotakmahindra</customer_card_issuer_bank>

  <customer_card_name>Visa</customer_card_name>
```

```
<customer_card_no_last4digits>2346</customer_card_no_last4digits>
<customer_id>14</customer_id>
<customer_pay_opt_type>OPTCRDC</customer_pay_opt_type>
<error_code></error_code>
<error_desc></error_desc>
<status>0</status>
</Add_Customer_Payment_Option_Result>
```

#### Failure:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Add_Customer_Payment_Option_Result>
  <customer_card_expiry>11/2015</customer_card_expiry>
  <customer_card_id></customer_card_id>
  <customer_card_issuer_bank>Kotakmahindra</customer_card_issuer_bank>

  <customer_card_name>Visa</customer_card_name>
  <customer_card_no_last4digits>2346</customer_card_no_last4digits>
  <customer_id>14</customer_id>
  <customer_pay_opt_type></customer_pay_opt_type>
  <error_code>51336</error_code>
  <error_desc>Customer Card Id: Invalid Parameter</error_desc>
  <status>1</status>
</Add_Customer_Payment_Option_Result>
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

#### Example JSON Response

##### Success:

```
{
  "customer_id": "14",
  "customer_card_id": "36",
  "customer_card_no_last4digits": "6545",
  "customer_card_expiry": "11/2015",
  "customer_pay_opt_type": "OPTCRDC",
  "customer_name_on_card": "subash yadav",
  "customer_card_name": "Visa",
  "customer_card_issuer_bank": "Kotakmahindra",
  "error_desc": "",
  "error_code": "", "status": 0
}
```

##### Failure:

```
{
  "customer_id":"14",
  "customer_card_id":"",
  "customer_card_no_last4digits":"2346",
  "customer_card_expiry":"11/2015",
  "customer_pay_opt_type":"",
  "customer_name_on_card":"subash yadav",
  "customer_card_name":"Visa",
  "customer_card_issuer_bank":"Kotakmahindra",
  "error_desc":"Customer Card Id: Invalid Parameter",
  "error_code":"51337",
  "status":1
}
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

### Example String Response

#### Format:

Status|customer\_id|customer\_card\_id|customer\_card\_no\_last4digits|customer\_card\_expiry|customer\_pay\_opt\_type|customer\_card\_name|customer\_card\_issuer\_bank|error\_code|error\_desc|

#### Failure:

**Example:**1|14||test user|1111|11/2015|OPTCRDC|Visa|Kotakmahindra|51337|Customer Id:Invalid parameters|

#### Success:

**Example:**0|14|415|test user|1111|11/2015|OPTCRDC|Visa|Kotakmahindra|||

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

## 4. Payouts Summary

Payouts Summary API call is used to list payouts summary for a merchant for a given settlement date.

### Request Parameters

Name	Description	Note
<b>enc_request</b> (required)	AES encrypted request data.	
<b>access_code</b> (required)	Unique CCAvenue access code which is generated when merchant registered their IP address. You must send this with each request.	
<b>request_type</b> (required)	API requests are accepted in XML, JSON or String. Specify the request type.	Possible value for request_type is “XML” or “JSON”
<b>response_type</b> (optional)	API returns responses in XML, JSON or String format. If left blank, the response will be in the same format as request.	Possible value for response_type is “XML” or “JSON”
<b>Command</b> (required)	Command value specifies the API calls. You must send this with each request.	Possible value for this API call is “ <b>payoutSummary</b> ”.
<b>settlement_date</b> (required)	Provide the settlement Date to find the payouts list.	Date must be in IST(dd-mm-yyyy)format.

### Example XML Request

```
<?xml version='1.0' encoding='UTF-8' standalone='yes'?>
  <payout_summary_query>
    <settlement_date>19-01-2016</settlement_date>
  </payout_summary_query>
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Example JSON Request

```
{
  'settlement_date': '19-01-2016'
}
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Response Parameters

Name	Description	Note
<b>status</b>	This states whether the call was successful or not. If value of this parameter is “1” then you need not decrypt the enc_response as it will contain plain error message.	Value “0” denotes that the API call was successful. Value “1” denotes API call failure. On enc_response is plain text representing the error message.
<b>enc_response</b>	AES encrypted response containing format as per <i>response_type</i> .	
<b>pay_amount</b>	Payout amount for a payout.	Decimal(12,2).
<b>pay_id</b>	Unique pay Id for payout.	Numeric (25).
<b>settlement_date</b>	Date of settlement.	Date in <b>IST(dd-mm-yyyy )</b> format.
<b>settlement_bank</b>	Name of bank in which settlement done.	String.
<b>settlement_currency</b>	Currency in which settlement is done.	String.  Examples: <b>INR – Indian Rupee</b>
<b>trans_currency</b>	Currency in which merchant processed the transaction.	String Examples: <b>INR – Indian Rupee</b> <b>USD – United States Dollar</b> <b>SGD – Singapore Dollar</b> <b>GBP – Pound Sterling</b> <b>EUR – Euro, official currency of</b>



		Eurozone
<b>sub_acc_id</b>	Unique sub account id for merchant if payouts done for sub account of merchant.	Alphanumeric with special characters (hyphen)(20).
<b>error_desc</b>	Reason if customer payment option is not going to be deleted successfully.	String Please refer <a href="#">below table</a> for failure message.
<b>error_code</b>	Error code for Failure reason.	String Please refer <a href="#">below table</a> for failure message.

### Example XML Response

#### Success:

```
<?xml version='1.0' encoding='UTF-8'?>

<Payout_Summary_Result>

    <error_code></error_code>

    <error_desc></error_desc>

    <pauyout_summary_list>

        <pauyout_summary_details
            pay_amount="211.76"
            pay_id="40907"
            settlement_date="19-01-2016"
            settlement_bank="KARUR VYSYA BANK"
            settlement_currency="INR" sub_acc_id="split-1"
            trans_currency="INR" />

    </pauyout_summary_list>

</Payout_Summary_Result>
```

#### Failure:

```
<?xml version='1.0' encoding='UTF-8'?>

<Payout_Summary_Result>
```

```
<error_code>51419</error_code>
```

```
<error_desc>Enc_request: No record found for given criteria.
```

```
</error_desc>
```

```
</Payout_Summary_Result>
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

### Example JSON Response

**Success:**

```
{  
  
  "Payout_Summary_Result":{  
  
    "error_code": "",  
  
    "error_desc": "",  
  
    "payout_summary_list":{  
  
      "payout_summary_details":[{  
  
        "settlement_bank": "KARUR VYSYA BANK",  
  
        "pay_amount": 211.76,  
  
        "trans_currency": "INR",  
  
        "pay_id": 40907,  
  
        "sub_acc_id": "split-1",  
  
        "settlement_date": "19-01-2016",  
  
        "settlement_currency": "INR"  
  
      ]  
  
    }  
  
  }
```

```
}
```

```
}
```

**Failure:**

```
{
```

```
  "Payout_Summary_Result":{
```

```
    "error_code":52014,
```

```
    "error_desc":"payout_date: Required parameter missing"
```

```
  }
```

```
}
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

## 5. PayId Details

PayId Details API call is used to list transactions for a given PayId.

### Request Parameters

Name	Description	Note
<b>enc_request</b> (required)	AES encrypted request data.	
<b>access_code</b> (required)	Unique CCAvenue access code which is generated when merchant registered their IP address. You must send this with each request.	
<b>request_type</b> (required)	API requests are accepted in XML, JSON or String. Specify the request type.	Possible value for request_type is "XML" or "JSON"
<b>response_type</b> (optional)	API returns responses in XML, JSON or String format. If left blank, the response will be in the same format as request.	Possible value for response_type is "XML" or "JSON"
<b>Command</b> (required)	Command value specifies the API calls. You must send this with each request.	Possible value for this API call is "payIdDetails".
<b>pay_id</b> (required)	Provide the settlement Date to find the payouts list.	Numeric(25).

### Example XML Request

```
<?xml version='1.0' encoding='UTF-8' standalone='yes'?>
```

```
<pay_id_details_query>
```

```
    <pay_id>40907</pay_id>
```

```
</pay_id_details_query>
```

**Note:** You will have to encrypt the above request and store in the "enc\_request" parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Example JSON Request

```
{
  'pay_id':'40907'
}
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Response Parameters

Name	Description	Note
<b>status</b>	This states whether the call was successful or not. If value of this parameter is “1” then you need not decrypt the enc_response as it will contain plain error message.	Value “0” denotes that the API call was successful. Value “1” denotes API call failure. On enc_response is plain text representing the error message.
<b>enc_response</b>	AES encrypted response containing format as per <i>response_type</i> .	
<b>pay_id</b>	Unique pay Id for payout.	Numeric (25).
<b>Amount</b>	Amount of transaction.	Decimal(12,2).
<b>amt_payable</b>	Payable amount for given transaction to merchant.	Decimal(12,2).
<b>bank_ref_no</b>	Bank reference number of aggregator bank.	varchar(25).
<b>bill_email</b>	Customer email id for the transaction.	Alphanumeric with special characters (hyphen, underscore, dot,@)(70).
<b>bill_name</b>	Order billing name for the order.	Possible value for name is Alphanumeric with special characters (space, hyphen,

		apostrophe, underscore, dot)(60).
<b>ccavenue_ref_no</b>	CCAvenue reference no allocated to the transaction.	Numeric(25).
<b>currency</b>	Currency in which merchant processed the transaction.	String Examples: <b>INR</b> – Indian Rupee <b>USD</b> – United States Dollar <b>SGD</b> – Singapore Dollar <b>GBP</b> – Pound Sterling <b>EUR</b> – Euro, official currency of Eurozone
<b>date_time</b>	This is the latest date and time when order status is modified.	DateTime in <b>IST( yyyy-MM-dd HH:mm:ss.SSS)</b> format.
<b>fees</b>	Fee applied to process transaction by CCAvenue.	Decimal(12,2).
<b>order_no</b>	Order No for the transaction.	AlphaNumeric with special characters(hyphen and underscore)(30).
<b>sub_acc_id</b>	Unique Id for sub Account of merchant if provided for transaction	Alphanumeric with special characters (hyphen)(20).
<b>tax</b>	Sum of Taxes applied for the transaction.	Decimal(12,2).
<b>txn_type</b>	Type of transaction included in payouts.	String Example: <b>CAPTURE</b>  <b>Chargeback</b>  <b>Dispute</b>  <b>MISC</b>  <b>Payout</b>  <b>REFUND</b>
<b>error_desc</b>	Reason if customer payment option is not going to be deleted successfully.	String. Please refer <a href="#">below table</a> for failure message.

<b>error_code</b>	Error code for the Failure reason.	String. Please refer <a href="#">below table</a> for failure message.
-------------------	------------------------------------	--

### Example XML Response

#### Success:

```
<?xml version='1.0' encoding='UTF-8'?>

<pay_id_details_Result>

    <error_code></error_code>

    <error_desc></error_desc>

    <pay_id>40907</pay_id>

    <pay_id_txn_details_list>

        <pay_id_txn_details

            amount="40.00"

            amt_payable="35.88"

            bank_ref_no="1452514407217"

            bill_email="chandrakant.patil@avenues.info"

            bill_name="Shashi"

            ccavenue_ref_no="205000170631"

            currency="INR"

            date_time="2016-01-11 17:42:58.223"

            fees="3.60"

            order_no="29649917"

            sub_acc_id=""


```



```
tax="0.52"

txn_type="Chargeback" />

</pay_id_txn_details_list>

</pay_id_details_Result>

Failure:
<?xml version='1.0' encoding='UTF-8'?>

<pay_id_details_Result>

    <error_code>51419</error_code>

    <error_desc>Enc_request: No record found for given criteria.</error_desc>

    <pay_id>40907</pay_id>

</pay_id_details_Result>
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

### Example JSON Response

**Success:**

```
{

  "pay_id_details_Result":{

    "error_code": "",

    "error_desc": "",

    "pay_id": 40907,

    "pay_id_txn_details_list":{

      "pay_id_txn_details"[[

        "amt_payable": 35.88,
```



```

        "bill_email":"chandrakant.patil@avenues.info",
        "fees":3.6,
        "bill_name":"Shashi",
        "order_no":29649917,
        "currency":"INR",
        "amount":40,
        "tax":0.52,
        "ccavenue_ref_no":205000170631,
        "bank_ref_no":1452514407217,
        "date_time":"2016-01-11 17:42:58.223",
        "txn_type":"Chargeback",
        "sub_acc_id":""
    }}
}
}
}

```

**Failure:**

```

{
  "pay_id_details_Result":{
    "error_code":52017,
    "error_desc":"pay_id: Invalid Parameter"
  }
}

```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.



## 6. Create Split Payout

When transaction happens through the main merchant who in turn has to distribute the amount among multiple sub-accounts, split payout shall be used. In this way the payout will be directly performed to the sub-accounts.

**Note:**

1. Split payout shall be performed only for the confirmed/shipped transactions against the confirmed amount.
2. Split payout call once made cannot be modified. So please ensure that the details are accurate before making the call.

### Request Parameters

Name	Description	Note
<b>enc_request</b> (required)	AES encrypted string containing request parameters.	
<b>access_code</b> (required)	Unique CCAvenue access code which is generated when merchant registered their IP address. You must send this with each request.	
<b>request_type</b> (required)	API requests are accepted in XML, JSON or String. Specify the request type.	Possible value for request_type is "XML" or "JSON" or "STRING".
<b>response_type</b> (optional)	API returns responses in XML, JSON or String format. If left blank, the response will be in the same format as request.	Possible value for response_type is "XML" or "JSON" or "STRING".
<b>split_tdr_charge_type</b>	This decides which party should be charged for the TDR fees in case of split payouts.	Possible value for split_tdr_charge_type is 'M' or 'S' or 'A' where <b>M</b> means 'Merchant' <b>S</b> means Submerchant <b>A</b> means All
<b>Command</b> (required)	Command value specifies the API calls. You must send this with each request.	Possible value for this API call is "createSplitPayout".
<b>reference_no</b>	Unique CCAvenue reference	Numeric(25)

<i>(required)</i>	number for the transaction.	
<b>splitAmount</b> <i>(required)</i>	SplitAmount is part of Order amount which needs to be given to particular sub account.	Decimal(12,2)
<b>subAcclId</b> <i>(required)</i>	Unique Sub Account ID.	String
<b>merComm</b> <i>(required)</i>	This is the value of merchant's commission as defined in the split payout.	Decimal(12,2)

### Example XML Request

```
<?xml version='1.0' encoding='UTF-8' standalone='yes'?>
<Create_Split_Payout_Query>
  <reference_no>204000166782</reference_no>
  <split_tdr_charge_type>M</split_tdr_charge_type>
  <merComm>2.0</merComm>
  <split_data_list>
    <split_data splitAmount='40.76' subAcclId='2954-1'/>
    <split_data splitAmount='40.75' subAcclId='2954-2'/>
    <split_data splitAmount='40.75' subAcclId='2954-3'/>
  </split_data_list>
</Create_Split_Payout_Query>
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Example JSON Request

```
{
  'reference_no': '204000167798',
  'split_tdr_charge_type': 'M',
  'merComm': '2.0',
  'split_data_list': [
    {'splitAmount': '0.50', 'subAcclId': '2954-1'},
    {'splitAmount': '0.50', 'subAcclId': '2954-2'}
  ]
}
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

## Example STRING Request

### Format:

```
reference_no|split_tdr_charge_type|merComm|splitAmount$subAcclId#splitAmount$subAcclId#
splitAmount$subAcclId|
204000166782|M|2.0|0.50$2954-1#0.50$2954-2#0.50$2954-3|
```

### Example:

```
XXXXXXXXXXXXX|X|X.X|XX.XX$XXXX#XX.XX$XXXX#XX.XX$XXXX|
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

## Response Parameters

Name	Description	Note
status	This states whether the call was successful or not. If value of this parameter is “1” then you need not decrypt the enc_response as it will contain plain error message.	Value “0” denotes the API call success. Value “1” denotes API call failure. On enc_response is plain text represents the error message.
enc_response	AES encrypted response containing format as per response_type	
error_code	Error code for API failure.	Numeric
reference_no	This is the reference number merchant sent in confirm request.	Numeric
error_desc	Failure reason for reference_no	String

error_code	error_desc
52001	Error while creating the split payout request.
52002	Split data is mandatory.
52003	Split amount is mandatory.
52004	Sub account id is mandatory.
52005	Split commission is mandatory.
52006	Split record already exists.
52007	Sum of split amount is not equal to order amount.
52008	Record not found.

52009	Commission should be numeric.
52010	split_tdr_charge_type is mandatory.
52011	Invalid split_tdr_charge_type.
52012	Split payout is not applicable.

### Example XML Response

#### Success:

```
<?xml version='1.0' encoding='UTF-8'?>
<Create_Split_Payout_Result>
  <error_code></error_code>
  <error_desc></error_desc>
  <reference_no>204000167800</reference_no>
  <status>0</status>
</Create_Split_Payout_Result>
```

#### Failure:

```
<?xml version='1.0' encoding='UTF-8'?>
<Create_Split_Payout_Result>
  <error_code>51015</error_code>
  <error_desc>Amount must be Numeric.</error_desc>
  <reference_no>204000167800</reference_no>
  <status>1</status>
</Create_Split_Payout_Result>
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

### Example JSON Response

#### Success:

```
{
  "Create_Split_Payout_Result":
  {
    "status":0,
    "error_code": "",
    "reference_no":204000167801,
    "error_desc":""
  }
}
```

#### Failure:

```
{
```

```
"Create_Split_Payout_Result":
{
  "status":1,
  "error_code":51015,
  "reference_no":204000167801,
  "error_desc":"Amount must be numeric."
}
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

### Example STRING Response

#### Success:

**Format:** reference\_no|status|error\_code|error\_desc

**Example:** 204000167802|0||

#### Failure:

**Format:** reference\_no|status|error\_code|error\_desc

**Example:** 204000167802|1|51015|Amount must be Numeric.

## 7. Split Refund

Split refund call is performed particularly for transactions which are processed through Split Payout.

**Note:**

1. Split refund call can be made only once against a transaction (either full or partial).
2. Multiple refunds for the same transaction are not possible as of now.
3. Though a split order is shipped, Split refund cannot be called until Split Payout is performed on that transaction.

### Request Parameters

Name	Description	Note
<b>enc_request</b> (required)	AES encrypted request data.	
<b>access_code</b> (required)	Unique CCAvenue access code which is generated when merchant registered their IP address. You must send this with each request.	
<b>request_type</b> (required)	API requests are accepted in XML, JSON or String. Specify the request type.	Possible value for request_type is "XML" or "JSON"
<b>response_type</b> (optional)	API returns responses in XML, JSON or String format. If left blank, the response will be in the same format as request.	Possible value for response_type is "XML" or "JSON"
<b>Command</b> (required)	Command value specifies the API calls. You must send this with each request.	Possible value for this API call is "splitRefund".
<b>reference_no</b> (required)	CCAvenue reference no allocated to the transaction.	Numeric(25).
<b>refundAmount</b> (required)	Transaction amount to be refunded. Amount can be full or partial of the transaction amount.	Decimal(12,2).
<b>subAcclId</b> (optional)	Unique Id for sub Account of merchant provided for split payout for which amount to be refunded.	Alphanumeric with special characters (hyphen)(20).
<b>refundRefNo</b> (required)	Unique reference number shared by merchant to refund the captured transaction amount.	AlphaNumeric(30)





### Example XML Request

```
<?xml version='1.0' encoding='UTF-8' standalone='yes'?>

    <split_refund_query>

        <reference_no>205000175099</reference_no>

        <split_data_list>

            <split_data refundAmount='40.00' subAcclId='2954-2'

                refundRefNo='45454' />

            <split_data refundAmount='40.00' subAcclId='2954-1'

                refundRefNo='6321' />

            <split_data refundAmount='20' refundRefNo='7787445' />

        </split_data_list>

    </split_refund_query>
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Example JSON Request

```
{'reference_no': '205000175099', 'split_data_list': [{ 'refundAmount': '40.00', 'subAcclId': '2954-1', 'refundRefNo': '45454' }, { 'refundAmount': '40.00', 'subAcclId': '2954-2', 'refundRefNo': '45454' }, { 'refundAmount': '20.00', 'refundRefNo': '454254' } ] }
```

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

## Response Parameters

Name	Description	Note
<b>status</b>	This states whether the call was successful or not. If value of this parameter is "1" then you need not decrypt the enc_response as it will contain plain error message.	Value "0" denotes that the API call was successful. Value "1" denotes API call failure. On enc_response is plain text representing the error message.
<b>enc_response</b>	AES encrypted response containing format as per <i>response_type</i> .	
<b>refund_status</b>	Refund status specifies whether given refund request is going to succeed or fail.	Numeric. Value "0" denotes refund was successful Value "1" denotes refund was failure.
<b>reason</b>	Reason if customer payment option is not going to be deleted successfully.	String. Please refer <a href="#">below table</a> for failure message.
<b>error_code</b>	Error code for Failure reason.	String Please refer <a href="#">below table</a> for failure message.

## Example XML Response

### Success:

```
<?xml version='1.0' encoding='UTF-8'?>

    <split_refund_result error_code="" reason=""

        refund_status="0" />
```

### Failure:

```
<?xml version='1.0' encoding='UTF-8'?>

    <split_refund_result error_code="52020"

        reason="Refund limit exceeds" refund_status="1" />
```



**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

### Example JSON Response

#### Success:

```
{"split_refund_result":{"reason":"","error_code":"","refund_status":0}}
```

#### Failure:

```
{"split_refund_result":{"reason":"Amount: Required parameter missing","error_code":51014,"refund_status":1}}
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.

## 8. Bin Details

Bin Details call is performed to fetch the card details of bin.

### Request Parameters

Name	Description	Note
<b>enc_request</b> (required)	AES encrypted request data.	
<b>access_code</b> (required)	Unique CCAvenue access code which is generated when merchant registered their IP address. You must send this with each request.	
<b>request_type</b> (required)	API requests are accepted in XML, JSON or String. Specify the request type.	Possible value for request_type is "XML" or "JSON" or "STRING"
<b>response_type</b> (optional)	API returns responses in XML, JSON or String format. If left blank, the response will be in the same format as the request.	Possible value for response_type is "XML" or "JSON" or "STRING"
<b>Command</b> (required)	Command value specifies the API calls. You must send this with each request.	Possible value for this API call is "binDetails".
<b>bin_number</b> (required)	First 6 digits of card number (Bin number).	Numeric(6).

### Example XML Request

```
<bin_details_query>
<bin_number>464042</bin_number>
</bin_details_query>
```

**Note:** You will have to encrypt the above request and store in the "enc\_request" parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Example JSON Request

```
{'bin_number': '464042' }
```

**Note:** You will have to encrypt the above request and store in the "enc\_request" parameter before



sending it to CCAvenue. Kindly refer to the encryption section.

### Example STRING Request

**Format:** bin\_number|

**Example:** 464042|

**Note:** You will have to encrypt the above request and store in the “enc\_request” parameter before sending it to CCAvenue. Kindly refer to the encryption section.

### Response Parameters

Name	Description	Note
status	This states whether the call was successful or not. If the value of this parameter is “1” then you need not decrypt the enc_response as it will contain the plain error message.	Value “0” denotes that the API call was successful. Value “1” denotes API call failure. On enc_response is plain text representing the error message.
enc_response	AES encrypted response containing the format as per <i>response_type</i> .	
bin_number	Bin number that was sent in request.	Numeric(6).
card_name	Name of the card of the selected Bin number.	String Possible values, "Visa Debit Card",  "MasterCard Debit Card",  "Maestro Debit Card",  "Amex",  "Diners Club",  "DISCOVER",  "JCB",  "MasterCard",  "Visa"
card_type	Card type of the given bin number.	String Possible values:

		"OPTDBCRD", "OPTCRDC"
<b>payment_option</b>	Payment option for the card of the given bin number	String. Values : <b>OPTCRD</b> -credit card <b>OPTDBCRD</b> -debit card
<b>issuing_bank</b>	Name of the bank that issue the card of the given bin number	String
<b>country</b>	Country of the bank that issue the card of the given bin number  Note this parameter is available only in xml and JSON response.	String
<b>error_desc</b>	Reason if API call does not find the record based on the given search criteria.	String. Please refer <a href="#">below table</a> for failure message.
<b>error_code</b>	Error code for Failure.	String. Please refer <a href="#">below table</a> for failure message.

### Example XML Response

#### Success:

```
<?xml version='1.0' encoding='UTF-8'?>
```

```
<bin_details_Result>
```

```
    <bin_number>464042</bin_number>
```

```
    <card_name>Visa Debit Card</card_name>
```

```
    <card_type>DBCRD</card_type>
```

```
    <country>UNITED STATES</country>
```

```
    <error_code></error_code>
```

```
    <error_desc></error_desc>
```

```
    <issuing_bank>FLAGSTAR BANK FSB</issuing_bank>
```

```
    <payment_option>OPTDBCRD</payment_option>
```

```
</bin_details_Result>
```

**Failure:**

```
<?xml version='1.0' encoding='UTF-8'?>
```

```
<bin_details_Result>
```

```
    <error_code>52026</error_code>
```

```
    <error_desc>bin_number: Invalid Parameter</error_desc>
```

```
</bin_details_Result>
```

**Example JSON Response****Success:**

```
{
  "bin_details_Result": {
    "issuing_bank": "FLAGSTAR BANK FSB",
    "card_type": "DBCRD",
    "bin_number": "464042",
    "payment_option": "OPTDBCRD",
    "error_code": "UNITED STATES",
    "card_name": "Visa Debit Card",
    "country": "",
    "error_desc": ""
  }
}
```

**Failure:**

```
{
  "bin_details_Result": {
    "error_code": "52026",
    "error_desc": "bin_number: Invalid Parameter"
  }
}
```

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.



### Example STRING Response

#### Success:

**Format:** Bin\_number|Card\_name|Card\_type|Payment\_option|Issuing\_bank|Error\_code|Error\_desc

#### Example:

464042|Visa Debit Card|DBCRD|OPTDBCRD|FLAGSTAR BANK FSB||

#### Failure:

**Example:** ||||52026|bin\_number: Invalid Parameter

**Note:** You will have to decrypt the above response from “enc\_response” parameter. Kindly refer to the decryption section.



**Note:** - Error Message when API status is “1”.

Error_code	Short Description	Long Description	Message
51401	Missing Parameter	Request_type: Required Parameter is Missing	“Ensure that request_type parameter is not blank.”
51402	Missing Parameter	Command: Required parameter missing	“Ensure that Command parameter is not blank.”
51403	Missing Parameter	Access_code: Required Parameter is Missing	“Ensure that access_code parameter is not blank.”
51404	Invalid Parameter	Request Type: Invalid Parameter	“Ensure that Request Type parameter value is only XML/JSON/STRING.”
51405	Invalid Parameter	Response Type: Invalid Parameter	“Ensure that Response Type parameter value is only XML/JSON/STRING.”
51407	Invalid Parameter	Access_code: Invalid Parameter	“You are not allowed to perform this operation.”
51408	Missing Parameter	Enc_request: Required parameter missing	“Ensure that enc_request parameter is not blank.”
51410	Invalid Parameter	Command: Invalid Parameter	“Ensure that the command name is not invalid. Please refer API document for valid command.”
51411	Invalid Parameter	JSON request format: Invalid Parameter format	“Ensure that the JSON request format is not invalid. Please refer API document for the API call.”
51412	Invalid Parameter	XML request format: Invalid Parameter format	“Ensure that XML request format is not invalid. Please refer API document for the API call.”
51413	Invalid Parameter	STRING request format: Invalid Parameter format	“Ensure that STRING request format is not invalid. Please refer API document for the API call.”
51419	Invalid Parameters	Enc_request: No record found for given criteria.	“No records were found for given search criteria.”
51420	Invalid Parameters	Enc_request : Unable to process request	“Unable to process your request for the API call.”
51421	Invalid Parameter	API version: Invalid Parameter	“Please use only supported version for the API call.”
-1	Invalid Parameter	Enc_request: Invalid Request	“Ensure that Request parameter is not

			invalid."
--	--	--	-----------

**Note:** Failure Reason for API call

Error Code	Short Description	Long Description	Reason / Error_desc	Applicable to
51001	Missing Parameter	Reference Number: Required parameter missing	"Ensure that the Reference Number parameter is not blank."	Order Status
51002	Invalid Parameter	Reference Number: Invalid Parameter	"Ensure that the Reference Number parameter is numeric."	Order Status
51003	Invalid Parameter	Reference Number: Invalid Parameter	"Ensure that the Reference Number parameter does not exceed 25 characters."	Order Status
51004	Invalid Parameter	Reference number/Order number: Invalid Parameter	"Ensure that reference number/order number is provided."	Order Status
51308	Invalid Parameter	Order List: Invalid Parameter	"No record was found for the given criteria."	Order Status
51313	Invalid Parameter	Order List: Invalid Parameter	"No records found."	Order Status
52014	Missing Parameter	payout_date: Required parameter missing.	"Ensure that request parameter payout_date is not blank."	Payout summary
52015	Invalid Parameter	payout_date: Invalid Parameter.	"Ensure that Request parameter payout_date is not invalid."	Payout summary
52016	Missing Parameter	pay_id: Required parameter missing.	"Ensure that request parameter pay_id is not blank."	Payid Details
52017	Invalid Parameter	pay_id: Invalid Parameter.	"Ensure that Request parameter pay_id is not invalid."	Payid Details

52023	Invalid parameter	sub_acc_id: Invalid parameter	"Ensure that parameter sub_acc_id is not invalid."	Split refund
51001	Missing Parameter	Reference Number: Required parameter missing	"Ensure that the Reference Number parameter is not blank."	Split refund
51002	Invalid Parameter	Reference Number: Invalid Parameter	"Ensure that the Reference Number parameter is numeric."	Split refund
51003	Invalid Parameter	Reference Number: Invalid Parameter	"Ensure that the Reference Number parameter does not exceed 25 characters."	Split refund
52002	Missing data	Split data is mandatory.	Ensure that the split data is mentioned.	Split refund
51014	Missing Parameter	Amount: Required parameter missing	"Ensure that Amount parameter is not blank."	Split refund
51015	Invalid Parameter	Amount: Invalid Parameter	"Ensure that the Amount parameter is in Decimal."	Split refund
51085	Missing Parameter	Refund Reference Number: Required parameter missing	"Ensure that 'refund reference no' parameter is not blank."	Split refund
51086	Invalid Parameter	Refund Reference Number: Invalid Parameter	"Ensure that Only letters and numbers are provided for refund Reference Number parameter."	Split refund
51087	Invalid Parameter	Refund reference number: Invalid Parameter	"Ensure that Refund reference number parameter does not exceed 100 characters."	Split refund
52018	Invalid Parameter	Refund not allowed	Refund not allowed.	Split refund
52019	Invalid Parameter	Multiple refund not allowed	Multiple refund not allowed	Split refund
52020	Invalid Parameter	Refund limit exceeds	Refund limit exceeds	Split refund
52021	Invalid Parameter	This is a disputed order	This is a disputed order	Split refund
52022	Invalid	Refund Period	Refund Period exceeds	Split refund

	Parameter	exceeds		
52025	Missing data	bin_number: Required parameter missing	"Ensure that Request parameter bin_number is not blank."	Bin Details
52026	Invalid Parameter	bin_number: Invalid Parameter	"Ensure that Request parameter bin_number is not invalid."	Bin Details