



## Blazecards Integration Document

Sandbox URL for Testing : <https://blazecardsbox.citruspay.com/cards-gateway/>

### Make Payment Call : Javascript Integration

It is the javascript intergration where post call is made by an ajax request which will be received by Citrus and in the response Citrus will send an HTML string. This HTML string needs to be appended to the body of calling HTML and post redirection to the 3D secure page.

**NOTE: Appending string and redirection script will be provided.**

<b>URL</b>	<b><code>https://&lt;citrus-base-url&gt;/rest/cardspg/mpi</code></b>
<b>Request Type</b>	<b>POST</b>

Eg : <https://blazecardsbox.citruspay.com/cards-gateway/rest/cardspg/mpi> [For sandbox testing]

### Expected Parameters for the call:

Parameter Name	Parameter Value	Mandatory
cardNo	Card Number of Customer	Yes
expiry	Expiry Date of Card(MM/YYYY)	Yes (not in case of Maestro but if value missing by default it will be set to 12/2049 , so need to be careful)
cvv	CVV of card	Yes (not in case of Maestro)
email	Email Id of Customer	Yes
phone	Phone Number of Customer	Yes
firstName	Customer First Name	No
lastName	Customer Last Name	No
amount	Transaction amount	Yes
Currency (ISO 4217 currency codes eg : INR for Indian Rupee)	Currency of transaction	Yes
returnUrl	URL to redirect on successful transaction	Yes
notifyUrl	URL to notify on backend	Yes
merchantTransactionId	Merchant Unique Transaction Id	No
merchantAccessKey	Merchant Acess Key Shared with Citrus	Yes
cardType (CREDIT or DEBIT)	Card Type Detail	Yes
cardScheme (Ref to Doc for card schemes)	Card Scheme Detail	Yes
customParameters	Custom Parameter Map	No



## Blazecards Integration Document

extraInfoMap	System Parameter	No
Signature (Refer to doc for signature generation code in java)	Signature Generated with Secret Key of Merchant	Yes

Signature Generation Sequence :

<email>+<phone>+<merchantTransactionId>+<amount>+<currency>+<returnUrl+<notifyUrl>

**NOTE: Use empty values in case value is null or empty.**

### Example JSON request for ajax Post :

```
{
  "merchantTransactionId" : "1446116348057",
  "merchantAccessKey" : "27AOYSJCQOR6VZ39V7JV",
  "amount": "1.00",
  "currency": "INR",
  "firstName": "test",
  "lastName": "test",
  "email": "test@gmail.com",
  "phone": "9028223363",
  "cardType": "CREDIT",
  "cardScheme": "VISA",
  "cardNo": "5326760128173213",
  "cvv": "986",
  "expiry": "11/2024",
  "returnUrl": "http://localhost:8080/local-Kit/jsp/TestCardsTxnResponse.jsp",
  "notifyUrl": "http://localhost:8080/local-Kit/jsp/TestNotifyResponse.jsp",
  "customParameters" : {
    "Alternate Phone No" : "8970605043",
    "Alternate Address" : "Pune"
  },
  "extraInfoMap" : : {
    "transaction System" : "website",
    "gateway name" : "Citrus"
  },
  "signature": "ab46ee09451f77073899920eeaec6cb9f2f5881c"
}
```

### Ajax post Example :

```
$.ajax({
  url : https://<citrus-base-url>/rest/cardspg/mpi,
  type : "POST",
  contentType: "application/json",
  data: JSON.stringify({ "merchantTransactionId" : "1446116348057",
    "merchantAccessKey" : "27AOYSJCQOR6VZ39V7JV",
    "amount": "1.00",
    "currency": "INR",
    "firstName": "test",
```



## Blazecards Integration Document

```
"lastName": "test",
"email": "test@gmail.com",
"phone": "9028223363",
"cardType": "CREDIT",
"cardScheme": "VISA",
"cardNo": "5326760128173213",
"cvv": "986",
"expiry": "11/2024",
"returnUrl": "http://localhost:8080/local-Kit/jsp/TestCardsTxnResponse.jsp",
"notifyUrl": "http://localhost:8080/local-Kit/jsp/TestNotifyResponse.jsp",
"customParameters" : {
    "Alternate Phone No" : "8970605043",
    "Alternate Address" : "Pune"
},
"extraInfoMap" : : {
    "transaction System" : "website",
    "gateway name" : "Citrus"
},
"signature": "ab46ee09451f77073899920eeaec6cb9f2f5881c")),
success : function(data) {
    /*
        check data.errorCode and if it is zero (0) and data.responseSignature
        matches merchant generated signature then only call redirectForm method
    */
    redirectForm(data);
}
});
}

function redirectForm(data){
    document.body.innerHTML +=data.info["citrusSubmitForm"];
    document.citrusSubmitForm.submit();
}
```

**Note : Always check data.errorCode and if it is zero (0) and data.responseSignature matches merchant generated signature then only call redirectForm method**

### Response ( ajax integration):

Between the MPI request sent and response recieved there is an internal process which takes care of ensuring that MPI call doesnt fail by retrying failed request from one PG to other configured MPI gateway.

### Success :

errorCode	Error Code of Flow (Refer Doc for Error Codes)
errorMessage	Error Message
responseSignature	Signature of Response

### Signature Generation Sequence :

Please verify the signature received from the gateway with your own generated signature.  
<errorCode>+<errorMessage>



## Blazecards Integration Document

### Redirection to 3D Secure / OTP page

On providing proper authentication of 3D Secure/ OTP page (providing 3D password or OTP whichever is applicable) user is redirected to payment capture stage.

### Failure :

Redirected to returnUrl with non-zero errorCode and errorMessage.

errorCode	Non zero value (Refer Doc for Error Codes)
errorMessage	Error message
responseSignature	Signature of response

### Signature Generation Sequence :

Please verify the signature received from the gateway with your own generated signature.

<errorCode>+<errorMessage>

### Response of successful payment :

Redirection to “returnUrl” as provided in request

Parameter Name	Value
errorCode	0
errorMessage	SUCCESS
merchantTransactionId	Provided merchant transaction id
amount	Amount of transaction
currency	Currency of transaction
pgTransactionStatus	Transaction status at PG side
mappedPgTransactionStatus	Mapped interpretation of transaction status at PG side
citrusTransactionId	Citrus transaction Id (important for further refund or enquiry)
responseSignature	Signature of the response

### Signature Generation Sequence :

Please verify the signature received from the gateway with your own generated signature.

<errorCode>+<errorMessage>+<citrusTransactionId>+<amount>+<currency(in upper case)>+<pgTransactionStatus>+<mappedPgTransactionStatus>



## Blazecards Integration Document

### NOTE :

Always make a check on **mappedPgTransactionStatus** when response received as **errorCode** and **errorMessage** tells us only that flow completed successfully.

mappedPgTransactionStatus values	Meaning
INITIALIZED	Transaction Initiated but before hitting 3D-Secure
PENDING_WITH_BANK	Transaction under control with bank
SUCCESS	Transaction Successfully Completed. Payment was successful.
FAIL	Transaction Failed. Payment unsuccessful.

### Response of Failure payment:

Redirection to “returnUrl” as provided in request

errorMessage	Non zero value (Refer Doc for Error Codes)
errorMessage	Error Message
responseSignature	Signature of the response

**Note :** There is a possibility of getting success response (response code 0) when card is refused of payment. So always check the transaction message for confirmation in case of success response.



## Blazecards Integration Document

### Cancel Transaction :

Cancel ongoing transaction at 3D secure page :

Parameter Name	Parameter Value	Mandatory
merchantTransactionId	Merchant Unique Transaction Id	No
merchantAccessKey	Merchant Access Key Shared with Citrus	Yes
citrusTransactionId	Transaction Id Citrus Shared with merchant in response of transaction at MPI call before redirection to 3D secure.	Yes
Signature (Refer to doc for signature generation code in java)	Signature Generated with Secret Key of Merchant	Yes

Signature Generation Sequence :

```
"merchantAccessKey="<merchantAccessKey>  
+ "&transactionId="<merchantTransactionId>  
+ "&amount="<amount>
```

**NOTE: Use empty values in case value is null or empty.**

### Sample Request:

```
{  
    "merchantTransactionId": "1446116348057",  
    "merchantAccessKey": "27AOYSJCQOR6VZ39V7JV",  
    "citrusTransactionId": "CTX20160627072703578UXL",  
    "signature": "ab46ee09451f77073899920eeaec6cb9f2f5881c"  
}
```

### Response :

#### Success :

Parameter Name	Parameter Value
system	BLAZECARD
errorCode	0
errorMessage	SUCCESS
citrusTransactionId	Citrus Transaction Id Provided by gateway at the time of transaction.
merchantTransactionId	Merchant Transaction Id Provided at the time of transaction



## Blazecards Integration Document

amount	Transaction Amount
currency	Transaction Currency
pgTransactionStatus	Actual transaction status with bank
mappedPgTransactionStatus	Mapped equivalent transaction status with bank
paymentMode	Payment mode of Transaction
authIdCode	Auth Id of transaction
issuerRefCode	Issuer Reference Code
firstname	Firstname as provided at the time of Request
lastName	Lastname as provided at the time of Request
pgRespCode	Pg Response Code
addressZip	Address zip as provided at the time of transaction
customParameters	Custom parameters provided at the time of transaction.
info	Detailed info of transaction (it will also contain few parameters required for signature generation)
signature	Signature generated for response

### Response Signature Sequence :

<merchantTransactionId>+<mappedTransactionStatus>+<amount>+<pgTxnNo>+<issuerRefNo>+<authIdCode>+<firstName>+<lastName>+<pgRespCode>+<addressZip>

**NOTE : Use “NA” in case if parameter is null or empty**

pgRespCode and pgTxnNo can be received from info of response.

E.g : data.info[“pgTxnNo”]



## Blazecards Integration Document

### Make Enquiry :

#### On Single Transaction Id :

It is the REST call with post request :

URL :	https://<citrus-base-url>/rest/cardspg/enquiryPayment
Request Type	POST

Eg : https://blazecardsbox.citruspay.com/cards-gateway/rest/cardspg/enquiryPayment [For sandbox testing]

### Expected Parameters :

Parameter Name	Parameter Value	Mandatory
citrusTransactionId	Transaction Id Citrus Shared with merchant in response of transaction	Yes
merchantAccessKey	Merchant access Key	Yes
signature	Signature	Yes

### Sample Request:

```
{
  "citrusTransactionId" : "CTX20151013072344P9RGGW",
  "merchantAccessKey" : "27AOYSJCQOR6VZ39V7JV",
  "signature" : "375aa77d4d1af090ddebf56074fe9438ebe6c54"
}
```

### Signature Generation Sequence :

<citrusTransactionId>

Eg: CTX20151013072344P9RGGW (based on above sample request)





## Blazecards Integration Document

### Response :

#### Success :

Parameter Name	Parameter Value
system	BLAZECARD
errorCode	0
errorMessage	SUCCESS
pgTransactionStatus	Actual transaction status with bank
mappedPgTransactionStatus	Mapped equivalent transaction status with bank
merchantTransactionId	Merchant Transaction Id Provided at the time of transaction
citrusTransactionId	Citrus Transaction Id Provided by gateway at the time of transaction.
info	Contains detailed Info of Enquiry

#### “info” Field Details :

Info contains the JSON object with key identifies the object as CitrusTransactionId :

It will also contain :

transactionDate  
citrusTransactionId  
merchantTransactionId  
pgTransactionId  
pgName  
transactionAmount  
transactionCurrency  
transactionStatus  
mappedPgTransactionStatus  
pgName  
transactionType

originalTransactionId (null in case if it is itself an original transaction)

transactionMessage (will contain actual message detailed message about transaction, if transaction completed successfully then it will return COMPLETE else will return some other detailed message)

extraInfoDetails (extra information related to transaction)



## Blazecards Integration Document

### Sample “response” Object:

```
{
  "system": "BLAZECARD",
  "errorCode": 0,
  "errorMessage": "SUCCESS",
  "merchantTransactionId": "1462432127349",
  "citrusTransactionId": "CTX20160505145849F7RY9X",
  "pgTransactionStatus": "AUTHORISED",
  "mappedPgTransactionStatus": "SUCCESS",
  "info": {
    "CTX20160505145849F7RY9X": {
      "transactionDate": "2016-05-05 14:58:50",
      "citrusTransactionId": "CTX20160505145849F7RY9X",
      "merchantTransactionId": "1462432127349",
      "pgTransactionId": "GX93MO",
      "pgName": "Yes Bank PG PAYZEN (Citrus Plus)",
      "transactionAmount": "1.00",
      "transactionCurrency": "INR",
      "transactionStatus": "COMPLETE",
      "mappedPgTransactionStatus": "SUCCESS",
      "transactionType": "SALE",
      "originalTransactionId": null,
      "transactionMessage": "Transaction Completed",
      "extraInfoDetails": {
        "remote_addr": "127.0.0.1",
        "firstname": "test",
        "addressZip": "null",
        "cardType": "DEBIT",
        "mobileNo": "9007588374",
        "operating_system_name": "Linux",
        "cardNo": "436303XXXXXX8874",
        "device_category": "PERSONAL_COMPUTER",
        "callType": "REST",
        "lastname": "test",
        "remote_host": "127.0.0.1",
        "operating_system_family": "Linux",
        "extTransId": "612614916147",
        "pg": "PAYZEN",
        "browser": "Chrome",
        "email": "test@gmail.com"
      }
    }
  }
}
```



## Blazecards Integration Document

### Failure :

system	BLAZECARD
errorCode	Non zero value (Refer Doc for Error Codes)
errorMessage	Error Message
mappedPgTransactionStatus	Equivalent transaction status with bank
merchantTransactionId	Merchant Transaction Id Provided at the time of transaction
citrusTransactionId	Citrus Transaction Id Provided by gateway at the time of transaction.



## Blazecards Integration Document

### Make Refund :

#### Single Transaction Refund :

It is the REST call with post request :

<b>URL</b>	<b>https://&lt;citrus-base-url&gt;/rest/cardspg/refundPayment</b>
<b>Request Type</b>	<b>POST</b>

Eg : <https://blazecardsbox.citruspay.com/cards-gateway/rest/cardspg/refundPayment> [For sandbox testing]

#### Expected Parameters :

Parameter Name	Parameter Value	Mandatory
citrusTransactionId	Citrus Transaction Id	Yes
amount	Amount to be refunded	Yes
currency	Currency Code	Yes
merchantAccessKey	Merchant Access Key	Yes
signature	Signature	Yes

#### Sample Request :

```
{
  "citrusTransactionId" : "CTX20151013072344P9RGGW",
  "amount" : "1.00",
  "currency" : "INR",
  "merchantAccessKey" : "27AOYSJCQOR6VZ39V7JV",
  "signature" : "375aa77d4d1af090ddebf56074fe9438ebe6c54"
}
```

#### Signature Generation Sequence :

<citrusTransactionId>+<amount>+<currency>

Eg: CTX20151013072344P9RGGW1.00INR (based on above sample request)



## Blazecards Integration Document

### Response :

#### Success :

Parameter Name	Parameter Value
system	BLAZECARD
errorCode	0
errorMessage	SUCCESS
merchantTransactionId	Merchant Transaction Id
pgTransactionStatus	Transaction Status with Bank
mappedPgTransactionStatus	Mapped equivalent status with bank
citrusTransactionId	New Citrus transaction Id for refund
Info	Contains Detailed info of Refund

#### “info” Field Details :

Info contains JSON objects with key identifier of object as CitrusTransactionId (value) :

Which will contain :

- transactionDate
- citrusTransactionId
- merchantTransactionId
- pgTransactionId
- pgName
- transactionAmount
- transactionCurrency
- transactionStatus
- transactionType
- mappedPgTransactionStatus
- originalTransactionId (Contains the original citrus transaction id)
- transactionMessage (It will contain actual message detailed message about transaction, if transaction completed successfully then it will return COMPLETE else will return some other detailed message)

and as “responseSignature”

please verify the response signature with the generated signature at merchant end :

#### Signature Generation Sequence :

<errorCode>+<errorMessage>+<citrusTransactionId>+<merchantTransactionId>+<pgTransactionStatus>

#### Sample “response” object :

```
{
  "system": "BLAZECARD",
  "errorCode": 0,
  "errorMessage": "SUCCESS",
```



## Blazecards Integration Document

```
"merchantTransactionId": "1462432127349",
"citrusTransactionId": "CTX20160505155044N7HDX4",
"pgTransactionStatus": "REFUND_SUCCESS",
"mappedPgTransactionStatus": "REFUND_SUCCESS",
"info": {
  "responseSignature": "5224fd57efbd51eeb40bd9af79a3ac606e0c49",
  "CTX20160505155044N7HDX4": {
    "transactionDate": "2016-05-05 15:50:44",
    "citrusTransactionId": "CTX20160505155044N7HDX4",
    "merchantTransactionId": "1462432127349",
    "pgTransactionId": "1LZWT6",
    "pgName": "Yes Bank PG PAYZEN (Citrus Plus)",
    "transactionAmount": "1.00",
    "transactionCurrency": "INR",
    "transactionStatus": "REFUND_COMPLETE",
    "mappedPgTransactionStatus": "REFUND_SUCCESS",
    "transactionType": "REFUND",
    "originalTransactionId": "CTX20160505145849F7RY9X",
    "transactionMessage": "Refund completed",
    "extraInfoDetails": {}
  }
}
```

### Failure :

errorCode	Non zero value (Refer Doc for Error Codes)
errorMessage	Error Message

### Note :

“mappedPgTransactionStatus” will show following status :

mappedPgTransactionStatus values	Meaning
REFUND_INITIALIZED	Refund Transaction Initiated.
REFUND_SUCCESS	Refund Transaction Successful.
REFUND_FAIL	Refund Transaction Failed.



## Blazecards Integration Document

### Supported Card Schemes:

Card Scheme	Accepted Input Value
MASTERCARD	MASTERCARD
VISA	VISA
MAESTRO	MAESTRO
RUPAY	RUPAY
DINERS	DINERS

### Signature Generation Logic (in Java)

data=<email>+<phone>+<merchantTransactionId>+<amount>+<currency>+<returnUrl+<notifyUrl>

```
javax.crypto.Mac mac = javax.crypto.Mac.getInstance("HmacSHA1");
mac.init(new javax.crypto.spec.SecretKeySpec(secret_key.getBytes(), "HmacSHA1"));
byte[] hexBytes = new org.apache.commons.codec.binary.Hex().encode(mac.doFinal(data.getBytes()));
String securitySignature = new String(hexBytes, "UTF-8");
```

### Error Response Codes :

Error Code	Error Message	Detailed Message
0	SUCCESS	SUCCESS
101	PARAMETER_MISSING	Mandatory parameter is missing
102	PARAMETER_VALIDATION_FAILED	Parameter validation is failed
103	<i>SIGNATURE_VALIDATION_FAILED</i>	Signature Validation failed
104	<i>TRANSACTION_NOT_FOUND</i>	Transaction is not found
105	<i>TRANSACTION_REFUSED</i>	Transaction refused by gateway
106	DENIED_BY_RISK	Transaction denied by Risk
107	INVALID_TRANSACTION_STATE	Transaction is in invalid state to process further
201	<i>GATEWAY_NOT_FOUND</i>	Gateway Unavailable
202	<i>MPI_CALL_FAILED</i>	Gateway operation failed
203	<i>GATEWAY_VALIDATION_FAILED</i>	Gateway validation failed
204	<i>GATEWAY_PAYMENT_PROCESS_FAILED</i>	Process payment failed
205	<i>INVALID_REQUEST_ID</i>	gateway denied
206	GATEWAY_RESPONSE_REQUEST_TIMED_OUT	Request Timed Out



## Blazecards Integration Document

	OUT	
207	UNEXPECTED_GATEWAY_RESPONSE	Unexpected response from Bank Gateway
208	USER_CANCELLED_TRANSACTION	3D verification failed or Declined by user
209	FRAUD_STATUS_FAILED	Blocked at ReD screening
210	RED_CONNECTION_FAILED	Cannot connect to ReD due to exception while initializing .ini file or encrypting card number
211	REFUND_FAILED	Refund Failed
212	CONCURRENT_TRANSACTION_MODIFICATION_ERROR	Concurrent transaction modified denied
301	WS_CLIENT_INITIALIZER_ERROR	connection not found
302	SIGNATURE_MISMATCH	data invalid
303	RESPONSE_SIGNATURE_VALIDATION_FAILED	data invalid
304	GATEWAY_DATE_ERROR	data invalid
305	GATEWAY_DATA_TAMPERED	Data Tampered in Between Channels
399	APPLICATION_ERROR	Internal Application error

### NOTE :

1. Under Exceptions Ranging between 101 to 199 will point to exceptions from merchant side.
2. Under Exceptions Ranging between 201 to 299 will point to exceptions from bank side.
3. Under Exceptions Ranging between 301 to 399 will point to exceptions from citrus side.