PayPal payments

Overview

Checkout can be used to accept PayPal payments.

This guide requires familiarity with Accepting payments.

PayPal processing fields

- customer - ID of a Customer created via the Customer API. A customer object can be created and attached to a Checkout. The customer object can store relevant customer details. Read all of the customer requirements needed for PayPal transactions.
- line_items - Shopping cart line items to be displayed on the page and sent to PayPal
- configurations.paypal - Object carrying the parameters required for making an Google Pay payment (**Required**)
    - payment_contract_id - This ID can be found in the portal or given to you by a Verifone employee. It is used to retrieve merchant's PayPal credentials necessary for a payment. (**Required**)
    - dynamic_descriptor - Short text to be displayed on the bank statement of the cardholder. Support varies per Payment Contract.
    - capture_now - Used for separate authorization and capture
    - invoice_number - PayPal invoice number
    - application_context - Additional merchant options to control the appearance of PayPal's landing page
    - description - Description of the payment shown to the shopper on the PayPal screen

Authorization and capture

PayPal transactions through the Checkout can be used to do a sale (capture_now = true) or to authorize without capturing immediately (capture_now = false). An authorized payment reserves the money and allows you to capture the funds at a later stage.

Handling responses

Whenever a card payment is processed via the Checkout, the responses events would contain additional fields specific to card payments in the **details** object.

Example of successful checkout via the Checkout:

```
[
{
"type": "TRANSACTION_INITIATED",
"id": "b5b5fa07-3c99-4613-92f7-e24c258b3269",
"timestamp": "2021-03-31T11:43:46.697Z",
"details": {
"id": "fd31890c-8c45-4846-a934-a7faa584e823",
"orderId": "71110696K0011224V",
"createdAt": "2021-03-31T11:43:45Z",
"status": "INITIATED",
"approvalUrl": "https://www.sandbox.paypal.com/checkoutnow?token=71110696K0011224V"
}
},
{
"type": "TRANSACTION_SUCCESS",
"id": "1aaebd63-2daf-4753-bc32-bcdb8c35b743",
"timestamp": "2021-03-31T11:44:07.717Z",
"details": {
```

```
"id": "fd31890c-8c45-4846-a934-a7faa584e823",
"captureId": "93459848W31137123",
"createdAt": "2021-03-31T11:44:06Z",
"isFinalCapture": true,
"payer": {
"payerId": "92FT3NN28KELQ",
"shippingAddress": {
"country": "IN",
"postalCode": "100047",
"countrySubdivision": "Maharashtra",
"city": "Mumbai",
"addressLine1": "addressLine1",
"addressLine2": "addressLine2",
}
},
"status": "AUTHORISED"
}
}
]
```

Example of failed PayPal payment via the Checkout:

```
[

{
"type": "TRANSACTION_FAILED",
"id": "f2041250-4fc2-4b3a-bc94-651ba099541a",
"timestamp": "2020-07-08T12:42:37.974Z",
"details": {
"id": "6d85d047-1e78-4f13-afe2-cc041bce524f",
"authorizationId": "4PB16132MH590854B",
"createdAt": "2021-01-06T04:40:30Z",
"expiresAt": "2021-02-04T04:40:30Z",
"status": "FAILED",
"payer": {
"payerId": "92FT3NN28KELQ",
"shippingAddress": {
"country": "IN",
"postalCode": "100047",
"countrySubdivision": "Maharashtra",
"city": "Mumbai",
"addressLine1": "addressLine1",
"addressLine2": "addressLine2",
}
}
}
}

]
```

- **Note**: To ensure that the redirection request was not tampered with, always check that the `transaction_id` received as query parameter in the redirection matches the `transaction_id` property of the retrieved Checkout. If those are not matching, this is indication of either an incorrect integration, that the redirection to your `return_url` did not originate from Verifone, or `transaction_id` was tampered with.
- You can now store the `transaction_id` value together with the order `1234` in your system to link the two together.

Scenarios

The table below describes the different outcomes of a Checkout. A full list error codes are available.

Description	Result	Merchant action
Failed transaction*	Redirect: `checkout_id={checkout_id} & transaction_id={transaction_id} & errorCode=123`	Unsuccessful payment (technical reason), do not display order confirmation
Successful transaction	Redirect: `checkout_id={checkout_id} & transaction_id={transaction_id}`	Display order confirmation
Customer visits the URL of an already completed Checkout	Redirect: `checkout_id={checkout_id} & errorCode=168`	Display corresponding message to customer. Checkout is completed whenever there was a single successful payment processed through it.
Customer visits the URL of an expired Checkout	Redirect: `checkout_id={checkout_id} & errorCode=169`	Display corresponding message to customer. Checkout is expired whenever the `expiry_time` is reached.
Customer visits the URL of a Checkout which has reached the maximum of failed payment attempts	Redirect: `checkout_id={checkout_id} & errorCode=165`	Display corresponding message to customer. Payments through a single Checkout can be attempted up to three times unsuccessfully.
Form validation errors / Other service failures on the Checkout page	Displays error alert to Customer on the page	Customer is prompted to correct their form input and retry the payment or try using alternate card or payment method

*** Failed transaction - Depending on which step in the payment process failed, the `transaction_id` might not always be present as the query parameter