

Overview

Submit Charge API allows you to submit a payment transaction to collect money from the customer. The API requires HMAC signature, please refer to *API Authentication* for more information about the signature.

API Endpoint	https://{server.com}/api/v3/charges
Method	POST
Content Type	Application-json

Payment Flow

1. The customer chooses *PIX* as a payment method, fills in the payment information, and submits the payment.
2. The merchant submits the payment request to PayAmigo platform
3. The PayAmigo platform process a payment transaction
4. Acquirer responses to PayAmigo successfully, the transaction is in charged status
5. The PayAmigo platform responds to the merchant, with transaction status "CHARGE_PENDING"
6. The PayAmigo platform will send a notification to the merchant after the payer completes the payment.

Refer to *Real-Time Notifications* for more information about the notification.

Request Parameters

Field	Required	Description
command	Y	Fixed as "CHARGE"
id	Y	1-100 chars, letter, number and hyphen are allowed. The unique transaction ID specified by the merchant. It fails if a 2nd charge request with the same transaction ID
customer.id	Y	1-100 chars, letter, number and hyphen are allowed. Customer id specified by the merchant. For a same customer please use the same ID, it helps PayAmigo to identify fraud
customer.firstName	Y	string of length 0 to 50. First name of customer
customer.lastName	Y	string of length 0 to 50. Last name of customer
customer.dateOfBirth	N	string of length 0 to 10. Customer date of birth in yyyy-mm-dd format
customer.ipAddress	Y	string of length 0 to 16. Customer IP address
lineItem.currency	Y	Currency ISO code. • Brazil: BRL
lineItem.amount	Y	The amount to collect
lineItem.description	Y	The product's description. 1-100 chars
payment.paymentMethod	Y	Fixed as "PIX" in uppercase.
payment.countryCode	Y	The ISO country code of the customer Brasil: BR
payment.accountHolder	Y	The customer's subscription ID.
payment.documentType	Y	String. Fixed as CPF for Brazil
payment.document	Y	String. CPF number
metadata.callbackURL	N	The callback URL to receive real-time notifications. If it is not present, notifications are sent to the url configured in merchant's profile

Sample Request

```
{
  "command": "CHARGE",
  "id": "202110092221",
  "customer": {
    "id": "TestUser",
    "firstName": "Jacob",
    "lastName": "Test",
    "dateOfBirth": "1987-11-23",
    "ipAddress": "127.0.0.1"
  },
  "lineItem": {
    "currency": "BRL",
    "amount": 12,
    "description": "5 Golden Rings"
  },
  "payment": {
    "paymentMethod": "PIX",
    "accountHolder": "Payer Name",
    "documentType": "CPF",
    "document": "67944528208",
    "countryCode": "BR"
  },
  "metadata": {
    "callbackURL": "https://merchant.com/listener/for/callback"
  }
}
```

Response Fields

To validate a payment is successful, you can follow these steps:

1. Check HTTP status, should be 201
2. Check the \$.status field in json response. Should be "CHARGED"

Field	Description
id	The unique transaction ID specified by the merchant.
internalId	The unique PayAmigo transaction ID generated by the system.
status	Transaction status, possible values are: CHARGE_PENDING -- the payment is submitted ERROR -- the payment is failed DECLINED -- the payment is declined by the gateway, it has failed.
merchantAccount	The merchant account name
lineItem	Echo the lineItem information submitted by the merchant
transactions	An array of transactions denoting the transaction activities
transactions[n].timestamp	The time of the activity performs, in UTC
transactions[n].	Historical details about this charge
transactions[n].gatewayProvidedId	The transaction ID on Acquirer side
transactions[n].type	Activity type, must be "CHARGE"
transactions[n].state	Activity status PENDING – the transaction is submitted APPROVED - the transaction is completed ERROR - the transaction is failed with error, please refer to 'resultCode' & 'responseText' for the error detail. DECLINED - the transaction is declined
transactions[n].amount	Activity amount
transactions[n].currency	Activity currency in ISO code format
transactions[n].resultCode	Activity result code Refer to " <i>Hard Decline & Result Codes</i> " for more detail
transactions[n].responseText	Description of the result code.

paymentType	It is "WALLET"
paymentMethod	Customer method of payment. PIX
created	Time this charge was first submitted in ISO-8601 format
paymentOption.type	Fixed as "LIGHT_BOX". Indicates that a PIX qr code should be displayed in lightbox.
paymentOption.uiType	Fixed as "IFRAME". Indicate that the off site has no constraints and should render properly regardless of if it is displayed in an iframe.
paymentOption.injectedHtml	Contains QR code
paymentOption.htmlCharset	Fixed as "UTF-8".

Sample Response for PIX.

Refer to \$.status, if it is "CHARGE_PENDING", show the QR code to the payer in a browser window. See injectedHtml section in the response.

```
{
  "id": "202110092221",
  "internalId": "dsfs6sd8fs6ds76fdsdd",
  "status": "CHARGED",
  "merchantAccount": "Demo_Merchant",
  "lineItem": {
    "currency": "BRL",
    "amount": 12,
    "description": "5 Golden Rings"
  },
  "transactions": [
    {
      "timestamp": "2021-10-09T08:47:29Z",
      "gatewayProvidedId": "02497b02-c03a-482e-82ce-75552eaec7ae",
      "type": "CHARGE",
      "state": "PENDING",
      "amount": 12,
      "currency": "BRL",
      "resultCode": 0,
      "responseText": "Success"
    }
  ],
  "paymentType": "WALLET",
  "paymentMethod": "PIX",
  "created": "2021-10-09T08:47:26Z",
  "paymentOption": {
    "type": "LIGHT_BOX",
    "uiType": "IFRAME",

```

```
    "injectedHtml": "<html><body><img src=...</body></html>",
    "htmlCharset": "UTF-8"
  }
}
```

Sample Error Response

If you receive \$.status = "ERROR" or "DECLINED", the payment is failed. Refer to transactions[0].responseText for the detailed error reason.

```
{
  "id": "202110092223",
  "internalId": "dsfs6sd8fs6ds76fdsdd",
  "status": "ERROR",
  "merchantAccount": "Demo_Merchant",
  "lineItem": {
    "amount": 12,
    "currency": "BRL",
    "description": "5 Golden Rings",
    "taxCode": "NT",
    "taxIncludedInPrice": true
  },
  "transactions": [
    {
      "timestamp": "2021-10-11T04:08:01Z",
      "type": "CHARGE",
      "state": "ERROR",
      "resultCode": 160,
      "responseText": "The following required inputs are missing or invalid:
countryCode"
    }
  ],
  "paymentType": "WALLET",
  "paymentMethod": "PIX",
  "created": "2021-10-11T04:08:00Z"
}
```

Sample Hard Declined Response

You receive the hard declined error for any integration error. HTTP status is not 2xx in this case.

```
{
  "requestId": "080bb40e-f3a6-484b-9a87-5b2ab45ad251",
  "errorCode": "entity_conflict_error",
  "message": "Data conflict: Duplicate entry '20210618001-202110092221' for key 'PURCHASE_RECORD_UK1'"
}
```

PIX Payment Expiration

PIX validity is generally set to 24 hours i.e. the user has 24 hours to complete PIX payment once QR code is generated. This expiry time can be adjusted based on the merchant's requirement. Once the PIX transaction has expired, the user is unable to complete the transaction and a new PIX transaction should be initiated. On expiration, the state of the transaction changes from CHARGE_PENDING to EXPIRED. Since the state changes in the system, a webhook/notification is sent back to the webhook URL. Below is a sample webhook notification for an EXPIRED transaction. The webhook handler can capture just the master “status”. The handler code does not need to sort through the transactions list to capture the latest “state”.

```
{
  "id": "7BRDA4E4ZEA1HVL",
  "internalId": "13047194823307264",
  "status": "EXPIRED",
  "merchantAccount": "Demo_merchant",
  "lineItem": {
    "amount": 1,
    "currency": "BRL",
    "description": "5 Golden Rings",
    "taxCode": "NT",
    "taxIncludedInPrice": true
  },
  "properties": {
    "QR_CODE_STR": "xxyy"
  },
  "transactions": [
    {
      "timestamp": "2025-02-26T15:56:38Z",
      "gatewayProvidedId": "265da948-4303-470b-8721-2bb0eb31a08b",
      "type": "CHARGE",
      "state": "PENDING",
      "amount": 1,
      "currency": "BRL",

```

```
"resultCode": 0,
"responseText": "Success."
},
{
  "timestamp": "2025-02-26T17:00:20Z",
  "gatewayProvidedId": "265da948-4303-470b-8721-2bb0eb31a08b",
  "type": "CHARGE",
  "state": "EXPIRED",
  "amount": 1,
  "currency": "BRL",
  "resultCode": 0,
  "responseText": "Success."
}
],
"paymentType": "WALLET",
"paymentMethod": "PIX",
"created": "2025-02-26T15:56:36Z",
"amountRemaining": 0
}
```