

Technical specifications on

Walletmix Payment Gateway Merchant Sandbox API

Walletmix



Walletmix Limited

House No: 2/B, Road No: 08,

Nikunja - 02,

Khilkhet,

Dhaka – 1229

Table of Contents

Document Control...	3
1. Objective of the document...	4
2. Basic Call Flow	4
3. Check Server Status	7
3.1 Get Server Details	7
4. Request For Payment	7
4.1 CURL Request	7
4.2 Request Parameters	8
4.3 Request Response	9
4.4 Forwarding URL...	9
5. Merchant Callback	10
6. Check Payment Status	10
6.1 CURL Request	10
6.2 Request Parameter	10
6.3 Response Fields	12
7. Transaction Status Codes...	13
8. Sample Source Code	13

9. Document Control

Date Modified	Version	Maintained and Developed by	Support Mail
14 Aug, 2022	2.7.1.22	Walletmix Software Team	support@walletmix.com

Walletmix

1. Objective of the document

This document will serve as a guide for developers/merchants who want to integrate Walletmix Payment Gateway for performing real time transaction verification against relevant services or banks. This will assist the merchants to prepare their infrastructure for integration with Walletmix Gateway.

2. Basic Call Flow

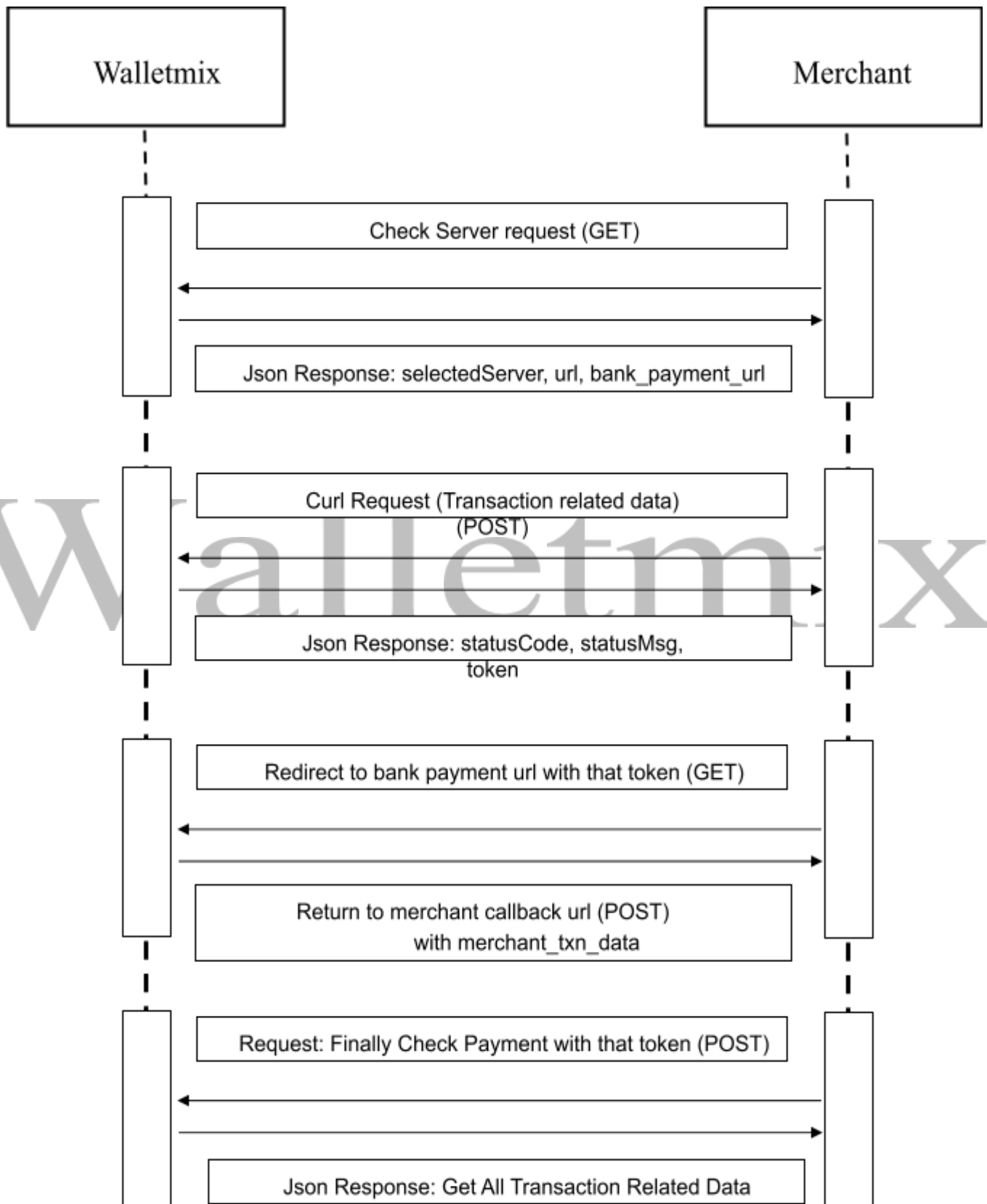
- To use the payment gateway service, merchants need to register Walletmix payment gateway service using a registration form. After successful registration, merchants will get a Merchant ID, Access App Key, Service access user name and password provided by Walletmix Limited.
- Customers will place an order like checkout from the merchant's website.
- All transaction related data will be stored in the merchant's database with an auto increment unique order_id field. Example: order_id = 105.
- Then merchants have to check server status to invoke “**check-server**”.
- Merchant will use the Walletmix Provided Merchant ID, Access App Key, Service access user name, password and that transaction related data to invoke “**init-payment-process**”. When invoke to “**init-payment-process**” Merchant will get a response of unique **token** against transaction related data after authentication & some validation; otherwise the merchant will get an error code and message.
- Merchant will forward a URL using GET method with that **token** and Customer will confirm the payment.
- After completing success or fail transaction it will forward to merchant callback URL using POST method with that token, transaction status and merchant order id.

- Don't trust these parameters. Merchant will have to again check transaction status to invoke “**check-payment**” with Merchant ID, Access App Key, Service access user name, password and that **token**.
- Then merchant will get a response of that transaction status
- Depending on the provided status can be updated in your database with that order_id.

Example: order_id = 105

Walletmix

The request/response call flow between Walletmix and merchant is shown in a diagram below:



3. Check Server Status

3.1 Get Server

Details Method: GET

Request URL: <https://sandbox.walletmix.com/check-server>

Response:

```
{
  "selectedServer": true,
  "url": "https://sandbox.walletmix.com/init-payment-process",
  "bank_payment_url": "https://sandbox.walletmix.com/bank-payment-process"
}
```

The screenshot displays a REST client interface with a GET request to `https://sandbox.walletmix.com/check-server`. The response is shown in JSON format, containing the following data:

KEY	VALUE	DESCRIPTION
Key	Value	Description

The response body is also shown in a code editor with the following JSON content:

```
1 {
2   "selectedServer": true,
3   "url": "https://sandbox.walletmix.com/init-payment-process",
4   "bank_payment_url": "https://sandbox.walletmix.com/bank-payment-process"
5 }
```

Additional interface details include tabs for Params, Authorization, Headers (10), Body, Pre-request Script, Tests, and Settings. The Body tab is active, showing the response in JSON format. The status bar indicates a 200 OK status, 222 ms time, and 1.12 KB size.

4. Request For Payment

4.1 CURL Request

Merchants send all transaction related data to walletmix using curl requests and generate a unique token against this data.

Method: POST

Request URL: Get From **3.1. Get Server Details**

4.2 Request Parameters

Parameters	Mandatory	Description	Sample Value
wmx_id	Yes	Merchant ID	WMX1234567891234
merchant_order_id	Yes	Merchant Order ID	105
merchant_ref_id	Yes	Merchant Reference ID	571607886240c
app_name	Yes	Only Domain Name	example.com
cart_info	Yes	Order Info	WMX1234567891234,http://example.com,MyApp
customer_name	Yes	Customer Name	Michel Schofield
customer_email	Yes	Customer Email	schofield@gmail.com
customer_add	Yes	Customer Address	House:01,Road:08,Nikunja-2,Dhaka-1229
customer_city	No	Customer City	Dhaka
customer_country	No	Customer Country	Bangladesh
customer_postcode	No	Customer Postcode	1229
customer_phone	Yes	Customer Phone	01910000000
shipping_name	No	Shipping Name	Linking Burrows
shipping_add	No	Shipping Address	1600 Amphitheater Pkwy, Mountain View
shipping_city	No	Shipping City	New York City
shipping_country	No	Shipping Country	United States
shipping_postCode	No	Shipping Postcode	1600
product_desc	Yes	Product Description	{2 X Adata 8GB Pendrive [800]=[1600]} {1 X A4Tech Mouse [700]=[700]} {shipping rate:40.00}-{discount amount:0.00}=2340.00
amount	Yes	Amount	2340
currency	Yes	Currency	BDT
extra_json	No	Extra Json Data	{"param_1":"data_1","param_2":"data_2","param_3":"data_3"}
		For EMI (Key:emi_period, value:3/6/9/12)	{"emi_period":"3"}
options	Yes	Options(base64_encode(s=example.com,i=192.168.100.105))	Cz26kHRwOi8vdGhlbWFSbGJKLmNvbS8saT0zNy4yMjEuMTk1LjE4Ng==
callback_url	Yes	Callback URL	http://example.com/callback.php
access_app_key	Yes	Access App Key	3dad54sf6sdf476crb64r274352348c5223452
authorization	Yes	Authorization	Basic 645623476crb64r274352348c52234==

Authorization key generate: 'Basic '.base64_encode(\$access_username.':'.\$access_password);

4.3 Request Response

```
{  
  "statusCode": "1000",  
  "statusMsg": "Success",  
  "token": "7341n23498jd923jd3487wjrc2t837487rcyqwerq9834"  
}
```

The screenshot displays a REST client interface with a POST request to `https://sandbox.walletmix.com/init-payment-process`. The request body is a JSON object with the following parameters:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> wmx_id	WMX54f[REDACTED]	
<input checked="" type="checkbox"/> merchant_order_id	13	
<input checked="" type="checkbox"/> merchant_ref_id	dagdfr3f4f97E	
<input checked="" type="checkbox"/> app_name	walletmix.com	
<input checked="" type="checkbox"/> cart_info	WMX54f[REDACTED]	
<input checked="" type="checkbox"/> customer_name	Tester	
<input checked="" type="checkbox"/> customer_email	k@markopolo.ai	
<input checked="" type="checkbox"/> customer_add	abcd avasdsgasvcas	
<input checked="" type="checkbox"/> customer_city	Dhaka	
<input checked="" type="checkbox"/> customer_country	Bangladesh	
<input checked="" type="checkbox"/> customer_postcode	1212	
<input checked="" type="checkbox"/> customer_phone	+8801794603050	
<input checked="" type="checkbox"/> product_desc	2 Facebook Dynamic Ads, 3 Facebook Static Ads, total 21 days...	

The response is a JSON object with the following structure:

```
{  
  "statusCode": "1000",  
  "statusMsg": "Success",  
  "token": "U17h5i1DWj[REDACTED]xSS"  
}
```

4.4. Forwarding URL:

After receiving a token from Walletmix Payment Gateway, Merchant will forward to the following URL using the get method. Then complete the payment process.

Method: GET

Request URL: <https://sandbox.walletmix.com/bank-payment-process/7341n23498jd923jd3487wjrc2t837487rcyqwerq9834>

5. Merchant Callback

After completing a successful or failed transaction it will make a post request to the merchant callback URL with “merchant_txn_data” key.

Method: POST.

Key: “merchant_txn_data” (Receive post data using this key)

Value:

```
{  
  "token": "7341n23498jd923jd3487wjrc2t837487rcyqwerq9834",  
  "txn_status": "1000",  
  "merchant_order_id": "105",  
}
```

6. Check Payment Status

6.1 CURL Request

Method: POST

Request URL: <https://sandbox.walletmix.com/check-payment>

6.2 Request Parameter

Field Name	Description	Sample Value
wmx_id	Merchant ID	WMX1234567891234
authorization	Authorization	Basic 645623476crb64r274352348c52234==
access_app_key	Access App Key	3dad54sf6sdf476crb64r274352348c5223452
token	Generated Token	7341n23498jd923jd3487wjrc2t837487rcyqwerq9834

Request

POST

https://sandbox.walletmix.com/check-payment

Send

Save

Params

Authorization

Headers (10)

Body

Pre-request Script

Tests

Settings

Cookies

Code

☐ none

☒ form-data

☐ x-www-form-urlencoded

☐ raw

☐ binary

☐ GraphQL

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> wmx_id	WMX54f1b4d3294d7			
<input checked="" type="checkbox"/> authorization	Basic d214XzE2NzY2Xzc0NjE6d214XzE2NzY2Xzl0MzE0			
<input checked="" type="checkbox"/> access_app_key	92dfc66dfde7fbc2cbb1debbb7dc64180379a165			
<input checked="" type="checkbox"/> token	yQbgatujKTPeMpsK4ZB820Cah3AQ3U7CDY41pHEWKdqH39F...			

Response:

```
1 {  
2   "wmx_id": "WMX-XXXXXXXXXX",  
3   "token": "f974422ff493XXXXXXXXXXXXadbl20a3XXXXXXXXXX2",  
4   "merchant_req_amount": "2430.00",  
5   "merchant_currency": "BDT",  
6   "merchant_amount_bdt": "2430.00",  
7   "conversion_rate": "1.00",  
8   "wmx_charge_bdt": "48.60",  
9   "service_ratio": "2.00",  
10  "bank_amount_bdt": "2478.60",  
11  "discount_bdt": "0.00",  
12  "merchant_order_id": "1",  
13  "request_ip": "110.44.0.138",  
14  "txn_status": null,  
15  "card_details": null,  
16  "is_foreign": 0,  
17  "payment_card": "QCASH-Qcash",  
18  "card_code": "8001",  
19  "payment_method": "5004",  
20  "init_time": "2016-10-30 17:21:48",  
21  "txn_time": "2016-10-30 17:28:05"  
22 }
```

6.3 Response Fields

Field Name	Description	Sample Value
wmx_id	Merchant ID	WMX1234567891234
token	Transaction Token	7341n23498jd923jd3487wjrc2t837487rcyqwerq9834
merchant_req_amount	Merchant Request Amount	200.00
merchant_currency	Merchant Currency	BDT/ USD/ EUR
merchant_amount_bdt	Merchant Amount BDT	200.00
conversion_rate	Currency Conversion Rate	0/100.00/ 200.00/ 300.00
wmx_charge_bdt	Walletmix Service Charge	0.00
service_ratio	Walletmix Service Ration	0.0
bank_amount_bdt	Bank Amount BDT	200.00
discount_bdt	Discount Amount	0.00
merchant_order_id	Merchant Order ID	105
request_ip	Request IP	192.168.0.100
txn_status	Transaction Status (1000 for success otherwise failed)	1000,1001,1009
is_foreign	Foreign Transaction Status (0 for Local 1 for foreign)	0,1
txn_details	Transaction Details	Send from Bank
card_details	Card Details	Visa /master/country ...
payment_card	Payment Card	BRAC-Visa
init_time	Initialize Time	2016-01-11 11:15:02
txn_time	Transaction Time	2016-01-11 11:17:02

7. Transaction Status Codes

txn_status	message	Interpretation
1000	Successful Transaction	Transaction SUCCESS
1001	Transaction Failure	Transaction REJECTED
1009	Transaction Failure	Transaction CANCELED
NULL	Transaction Failure	Transaction ATTEMPT

8. Sample Source Code

- ✓ Walletmix Class File
 - <http://pastebin.com/T6wBsEQU>
- ✓ Data Submission File
 - <http://pastebin.com/iee37Y98>
- ✓ Data Callback File
 - <http://pastebin.com/MwG0DDje>