

# DEVELOPER GUIDELINES API DOCUMENT GATEWAY

## 2025



AN ISO/IEC 27001:2022 CERTIFIED PAYMENT SERVICE  
OPERATOR LICENSED BY NEPAL RASTRA BANK



## Contents

1. Introduction .....	3
2. Authentication Header .....	3
3. Signature Generation .....	3
3.1 Example Signature Generation .....	4
4. Steps to Successfully Integrate Gateway .....	4
4.1 Get Payment Instrument Details .....	4
4.2 Get Service Charge .....	6
4.3 Get Process ID .....	7
4.4 Redirect to OnePG Gateway .....	9
4.5 Notification URL (Webhook Listener) .....	10
4.6 Response URL (Customer Redirection) .....	11
4.7 Check Transaction Status .....	12

## 1. Introduction

Nepal Payment Solution Ltd (NPS) is a licensed **Payment System Operator (PSO)** in Nepal, incorporated under the Company Act 2063 (with amendments) and authorized by Nepal Rastra Bank.

NPS provides a fast, secure, and developer-friendly API to enable online payments for merchants. The **OnePG Payment Switch** dynamically routes transactions between acquirers and payment service providers, improving transaction success rates while resolving connectivity and security challenges.

## 2. Authentication Header

OnePG uses **Basic Authentication** in HTTP headers.

- **Format:** `Authorization: Basic <Base64(apiusername:password)>`
- **Example:** For username `demo` and password `p@55w0rd`

```
Authorization: Basic ZGVtbzpwQDUldzByZA==
```

Include this header in all API requests.

## 3. Signature Generation

OnePG requires **HMAC-SHA512** signatures for message integrity.

Steps:

1. Alphabetically order the request payload keys.
2. Concatenate their values.
3. Apply HMAC-SHA512 with the **secret key**.
4. Convert the output to a **lowercase hexadecimal string**.

Example JSON payload:

```
{
  "MerchantId": "9",
  "MerchantName": "TestMerchant"
}
```

Concatenate values:

```
Value = MerchantId + MerchantName = "9TestMerchant"
```

Hash with key `"SecretKey"` → HMAC-SHA512 → hexadecimal string.

### 3.1 Example Signature Generation

*C#:*

```
string signature = HMACSHA512("9TestMerchant", "SecretKey");

internal static string HMACSHA512(string text, string secretKey)
{
    var hash = new StringBuilder();
    byte[] keyBytes = Encoding.UTF8.GetBytes(secretKey);
    byte[] textBytes = Encoding.UTF8.GetBytes(text);

    using (var hmac = new HMACSHA512(keyBytes))
    {
        byte[] hashValue = hmac.ComputeHash(textBytes);
        foreach (var b in hashValue)
            hash.Append(b.ToString("x2"));
    }

    return hash.ToString();
}
```

*Python:*

```
import hmac, hashlib

def generate_hmac_sha512(value: str, secret_key: str) -> str:
    hash_object = hmac.new(secret_key.encode(), value.encode(), hashlib.sha512)
    return hash_object.hexdigest()
```

*JavaScript (Node.js):*

```
const crypto = require('crypto');

const generateHMACSHA512 = (value, secretKey) => {
    const hmac = crypto.createHmac("sha512", secretKey);
    hmac.update(value, "utf8");
    return hmac.digest("hex");
};
```

*PHP:*

```
function generateHMACSHA512($plainText, $secretKey) {
    $hash = hash_hmac('sha512', $plainText, $secretKey, true);
    return strtolower(bin2hex($hash));
}
```

## 4. Steps to Successfully Integrate Gateway

### 4.1 Get Payment Instrument Details

Returns available payment instruments with metadata like name, code, institution, type, and logo.

- **URL:** <https://apisandbox.nepalpayment.com/GetPaymentInstrumentDetails>
- **Method:** POST

- **Content-Type:** application/json
- **Headers:** Basic Authentication

#### Request Parameters:

Field	Required	Description
MerchantId	Yes	Provided by OnePG
MerchantName	Yes	Provided by OnePG
Signature	Yes	HMAC-SHA512 of form values

#### Response:

Field	Description
Code	Returns either "0" or "1"
Message	Returns Success or Error
Errors	Returns List of error object in case of code 1
Data	Returns List of objects in case of code 0. Please see below table
Institution Name	Name of Institution
Instrument Name	Name of Instrument
Instrument Code	Unique Instrument Code (Can be used in <a href="#">Redirect To OnePG</a> )
Bank Type	Instrument Category

#### Request Example:

```
{
  "MerchantId": "6300",
  "MerchantName": "testapi",
  "Signature": "34bddc3a666d5bc1481e0ffe75e137dc..."
}
```

#### Response Example:

```
{
  "code": "0",
  "message": "Success",
}
```

```

"errors": [],
"data": [
  {
    "InstitutionName": "Card Checkout NIC Asia",
    "InstrumentName": "Card Checkout NIC Asia",
    "InstrumentCode": "NICCARD",
    "BankType": "checkoutcard",
    "LogoUrl":
"https://apisandbox.nepalpayment.com/UploadedImages/PaymentInstitution/LogoUrl-
202204081553S15.PNG"
  },
  {
    "InstitutionName": "Test Bank",
    "InstrumentName": "Test Bank",
    "InstrumentCode": "TMBANK",
    "BankType": "MBanking",
    "LogoUrl":
"https://apisandbox.nepalpayment.com/UploadedImages/PaymentInstitution/LogoUrl-
202104011205S44.png"
  }
]
}

```

## 4.2 Get Service Charge

Returns service charge based on amount and instrument.

- **URL:** <https://apisandbox.nepalpayment.com/GetServiceCharge>
- **Method:** POST
- **Content-Type:** application/json
- **Headers:** Basic Authentication

### Request Parameters:

Field	Required	Description
MerchantId	Yes	Provided by OnePG
MerchantName	Yes	Provided by OnePG
Amount	Yes	Transaction amount
InstrumentCode	Yes	Generated in get payment instruments details
Signature	Yes	HMAC-SHA512 of form values

#### Response:

Field Name	Description
<b>Code</b>	Returns either "0" or "1"
<b>message</b>	Returns Success or Error
<b>Errors</b>	Returns List of error object in case of code 1
<b>Data</b>	Returns an object containing Amount, CommissionType, ChargeValue and TotalChargeAmount in case of code 0 see below table

#### Request Example:

```
{
  "MerchantId": "6300",
  "MerchantName": "testapi",
  "Amount": "100",
  "InstrumentCode": "TMBANK",
  "Signature": "48e33b8e2d54d87ddaebea05395f09e..."
}
```

#### Response Example:

```
{
  "code": "0",
  "message": "Success",
  "errors": [],
  "data": {
    "Amount": "100",
    "CommissionType": "f",
    "ChargeValue": "5",
    "TotalChargeAmount": 5.0
  }
}
```

### 4.3 Get Process ID

Generates a unique token (ProcessId) for each transaction.

- **URL:** <https://apisandbox.nepalpayment.com/GetProcessId>
- **Method:** POST
- **Content-Type:** application/json
- **Headers:** Basic Authentication

### Request Parameters:

Field	Required	Description
MerchantId	Yes	Provided by OnePG
MerchantName	Yes	Provided by OnePG
Amount	Yes	String decimal amount value
MerchantTxnId	Yes	Unique Merchant Transaction Id Identifier
Signature	Yes	HMAC-SHA512 of form values

### Response:

Field Name	Description
Code	Returns either "0" or "1"
message	Returns Success or Error
Errors	Returns List of error object in case of code 1
Data	Returns an object containing ProcessId in case of code 0. Please see below table
Process Id	Unique Gateway Process Id (Token) identifier

### Request Example:

```
{
  "MerchantId": "6300",
  "MerchantName": "testapi",
  "Amount": "100",
  "MerchantTxnId": "Txn_01",
  "Signature": "365509156f54d9c7ef676b0dc52fa09..."
}
```

### Success Response:

```
{
  "code": "0",
  "message": "Process Id generated successfully",
  "errors": [],
  "data": {
    "ProcessId": "54D0A55C_4D9E_4EDC_A795_262101D09CF8"
  }
}
```



```
}
```

#### Error Response:

```
{
  "code": "1",
  "message": "Error",
  "errors": [
    {
      "error_code": "1",
      "error_message": "Duplicate Record"
    }
  ],
  "data": null
}
```

## 4.4 Redirect to OnePG Gateway

- Validates Process ID and Merchant Transaction ID.
- Directs the customer to the selected **InstrumentCode** or allows selection on the gateway page.
- **Signature and Auth header are not required** for redirection.

#### Form Submission Details:

- Action URL: <https://gatewaysandbox.nepalpayment.com/Payment/Index>
- Method: POST
- Content-Type: multipart/form-data

#### Request Parameters:

Field	Required	Description
MerchantId	Yes	Provided by OnePG
MerchantName	Yes	Provided by OnePG
Amount	Yes	Transaction amount
MerchantTxnId	Yes	Must match GetProcessId
TransactionRemarks	No	Optional for customer statement
InstrumentCode	No	Optional → direct redirection
ProcessId	Yes	Unique token from GetProcessId
ResponseUrl	No	Redirects to this URL after payment; defaults to URL set by NPS if not provided.

## Sample HTML Form:

```
<html>
<head>
  <title>Gateway Test</title>
</head>
<body onload="document.getElementById('form').submit()">
  <form method="post" action="https://gatewaysandbox.nepalpayment.com/Payment/Index"
  id="form">
    <input type="hidden" name="MerchantId" value="6300"/>
    <input type="hidden" name="MerchantName" value="testapi"/>
    <input type="hidden" name="MerchantTxnId" value="Txn_UAT01"/>
    <input type="hidden" name="Amount" value="100"/>
    <input type="hidden" name="ProcessId"
  value="0E92183C_015D_4D1A_8467_8F11DFB136E0"/>
    <input type="hidden" name="InstrumentCode" value=""/>
    <input type="hidden" name="TransactionRemarks" value="test checkout gateway"/>
    <input type="hidden" name="ResponseUrl" value="https://www.redirection.com"/>
  </form>
</body>
</html>
```

## Notes:

- **ResponseUrl** is optional in above form.
  - If provided, the customer will be redirected to the specified URL after payment completion.
  - If not provided, the system will redirect to the default URL configured in the NPS system.
- MerchantTxnId, Amount, and ProcessId must match the values returned from GetProcessId.

## 4.5 Notification URL (Webhook Listener)

Merchants must create a **Notification URL Listener** (webhook endpoint) on their server and provide this URL to the OnePG gateway team during setup.

- **Server-to-server GET request** after payment.
- **Parameters:** MerchantTxnId, GatewayTxnId.
- **Steps:**
  1. Accept GET requests without authentication blocking.
  2. Validate transaction via CheckTransactionStatus API.

3. Update backend transaction status.
4. Handle duplicate hits gracefully.

#### Response:

- "received" → first notification
- "already received" → duplicate

#### Example of Plaintext



OR

Already Received

#### Example Notification Request:

```
GET {MerchantNotificationURL}?MerchantTxnId=100900&GatewayTxnId=100000004593
```

## 4.6 Response URL (Customer Redirection)

Merchants should also create a **Response URL** page and provide it to the OnePG gateway team.

- **Client-facing page** to show customers the transaction receipt or confirmation message.
- Parameters sent via GET: MerchantTxnId, GatewayTxnId.
- Unlike the **Notification URL**, this URL is **not used for backend validation**.
- Always rely on the Notification URL to confirm the final transaction status.

#### Example Response Request:

```
GET {MerchantResponseURL}?MerchantTxnId=100900&GatewayTxnId=100000004593
```

## Example

<b>Gateway Transaction Id</b>
100000004593
<b>Transaction Id</b>
100900
<b>Amount</b>
100.00
<b>Bank Name</b>
Test Bank
<b>Payment Type</b>
Test EBanking

## Key Differences:

Feature	Notification URL (Webhook)	Response URL (Redirection)
<b>Call Type</b>	Server-to-server	Client browser
<b>Purpose</b>	Backend validation & status update	Display receipt / confirmation
<b>Accessibility</b>	Must remain open	Customer-facing
<b>HTTP Method</b>	GET	GET
<b>Mandatory</b>	Yes	Yes

## 4.7 Check Transaction Status

Verify the final transaction status.

- **URL:** <https://apisandbox.nepalpayment.com/CheckTransactionStatus>
- **Method:** POST
- **Content-Type:** application/json
- **Headers:** Basic Authentication

## Request Parameters:

Field	Required	Description
<b>MerchantId</b>	Yes	Provided by OnePG
<b>MerchantName</b>	Yes	Provided by OnePG
<b>MerchantTxnId</b>	Yes	Must match previous transaction or Notification URL
<b>Signature</b>	Yes	HMAC-SHA512 of request payload

### Response:

Field Name	Description
Code	Returns either "0" or "1"
Message	Returns Success or Error
Errors	Returns List of error object in case of code 1
Data	Returns an object containing Transaction detail case of code 0. Please see below table
ServiceCharge	Returns the service charge value of the transaction. If a <b>service charge is less than 0</b> then will return the service charge value as ". {decimal places}", <b>integer if no decimal places</b> , and returns the <b>decimal value for decimal places</b> .
TransactionRemarks	Remarks of the transaction
TransactionRemarks2	Remarks of the transaction
TransactionRemarks3	Remarks of the transaction
ProcessId	Unique Gateway Process Id (Token) identifier
TransactionDate	Date of transaction. Format yyyy-MM-dd HH:mm:ss
MerchantTxnId	Merchant Transaction Id Identifier
CbsMessage	Original message received from the bank or wallet system. This field can be <b>null</b> if no message is returned.
Status	CBS Transaction status. May return value either of below. <ul style="list-style-type: none"> <li>• Success</li> <li>• Fail</li> <li>• Pending</li> </ul>
Institution	Name of Institution
Instrument	Name of Instrument
PaymentCurrency	
ExchangeRate	

### Request Example:

```
{
  "MerchantId": "6300",
  "MerchantName": "testapi",
```

```

    "MerchantTxnId": "Txn_01",
    "Signature":
"80bceff8faf142677fc8c112b20f2cccae15cd6b140535d383c5aae76213b56f8a12f9905692030b9
99c07b7327c6b8195245e77db3aaffa44809f76ece36adc"
  }

```

## Response Example

### Success:

```

{
  "code": "0",
  "message": "Success",
  "errors": [],
  "data": {
    "GatewayReferenceNo": "100000035434",
    "Amount": "100",
    "ServiceCharge": "5",
    "TransactionRemarks": "test checkout gateway",
    "TransactionRemarks2": "",
    "TransactionRemarks3": "",
    "ProcessId": "0E92183C_015D_4D1A_8467_8F11DFB136E0",
    "TransactionDate": "2023-08-08 14:01:56",
    "MerchantTxnId": "Trnx UAT1235",
    "CbsMessage": "",
    "Status": "Success",
    "Institution": "Test Bank",
    "Instrument": "Test MBanking",
    "PaymentCurrency": "NPR",
    "ExchangeRate": "1"
  }
}

```

### Pending:

```

{
  "code": "0",
  "message": "Success",
  "errors": [],
  "data": {
    "GatewayReferenceNo": "100000071528",
    "Amount": "100",
    "ServiceCharge": "5",
    "TransactionRemarks": "test",
    "TransactionRemarks2": "tst remarks 2",
    "TransactionRemarks3": "",
    "ProcessId": "1277141C_9A5B_4CB7_A1C9_07CCE2C6F6E3",

```

```

    "TransactionDate": "2025-09-23 15:04:17",
    "MerchantTxnId": "M-TXN-XJZN7",
    "CbsMessage": "",
    "Status": "Pending",
    "Institution": "Test Bank",
    "Instrument": "Test MBanking"
  }
}
Failed:
{
  "code": "0",
  "message": "Success",
  "errors": [],
  "data": {
    "GatewayReferenceNo": "100000071472",
    "Amount": "10000",
    "ServiceCharge": "500",
    "TransactionRemarks": "test",
    "ProcessId": "F2D403A0_3A9A_4538_8EF2_777722D74505",
    "TransactionDate": "2025-09-21 17:38:48",
    "MerchantTxnId": "M-TXN-CWWVL",
    "CbsMessage": "Transaction failed",
    "Status": "Fail",
    "Institution": "Himalayan Bank",
    "Instrument": "HBL Card Checkout"
  }
}

```

### *Transaction Status Values:*

- Success → Transaction completed successfully
- Fail → Transaction failed
- Pending → Transaction is in progress

### **Error Example:**

```

{
  "code": "1",
  "message": "Error",
  "errors": [
    {
      "error_code": "3",
      "error_message": "Transaction not initialized yet"
    }
  ],
  "data": null
}

```