



V. 1.8 (S2S)

1. OVERVIEW

This document describes the steps for technical integration process between merchant website / application and Payfair Payments.

Through Payfair Payments, your customers can make electronic payments through various payment modes such as:

- Net banking
- UPI

Payfair Payments also offers you a business UI (<https://pay.payfair.in/>) where you have access to all your prior transaction/payment details, settlement details, analytics, etc.

You can also use this UI to create invoices singly or in bulk, set reminders, recurring billing, and many more features, manage your payables, vendor payments, set split ratios for vendor payments, process refunds, etc. This online interface can be accessed through <https://pay.payfair.in/>.

LIST OF CURRENCY CODES:

| Currency | Currency Code |
|----------|------------------------|
| INR | Indian Rupee |
| AED | UAE Dirham |
| EUR | Euro |
| GBP | Pound Sterling |
| USD | US Dollar |
| MUR | Mauritius Rupee |
| RWF | Rwanda Franc |
| LKR | Sri Lanka Rupee |
| XOF | West African CFA franc |

2. PAYMENT REQUEST API

When you integrate with Payfair Payments, the customer will be re-directed from your merchant website to the Payfair Payments payment page. After completion of the transaction, Payfair Payments will direct the customer back to the merchant website.

2.1. Steps for Integration

- You need to submit a **POST REQUEST** to our server, at the below mentioned URL

<https://pay.payfair.in/v2/paymentseamlessrequest>

Note: hash is a mandatory parameter. If your hash is not properly calculated or does not match for whatever reason, we will not be able to process the payment. The usage of hash is explained in subsequent sections.

- When you call this API, the customer is necessarily re-directed to Payfair payment page. After the customer makes the payment through Payfair Payments (entering his card details or netbanking details etc.), we direct the customer back to your merchant site.

Note: If you need the customer to enter credit card details on your (merchant) website and would NOT want us to redirect to the Payfair Payments page, we can get that done, provided you are PCI-DSS certified. If you are not certified and would like to get certified, let us know. We will guide you appropriately on how to get it done.

- We recommend that you check the hash at your end again, after we send back the response to you. This is essential to prevent user data tampering fraud.
- Transaction ID and order ID:
 - When you submit your transaction request to Payfair Payments, you need to submit an order ID as part of the request. This order ID can be used by you as a universal reference number for all transaction requests submitted by you.
 - When your customer clicks the "Pay" button on the payment page, a unique transaction ID is assigned to the transaction.
 - Order ID acts as a "merchant reference number". You must maintain uniqueness of your order IDs.

2.2. Parameters to be POSTed in Payment Request

URL: <https://pay.payfair.in/v2/paymentseamlessrequest>

| Parameter Name | Description | Data type | Optional / Mandatory |
|------------------|--|---------------|----------------------|
| api_key | Payfair Payments would assign a unique 36-digit merchant key to you. This key is exclusive to your business/login account. If you have multiple login accounts, there will necessarily be one different api_key per login account that is assigned to you. | varchar(36) | mandatory |
| order_id | This is your (merchant) reference number. It must be unique for every transaction. We do perform a validation at our end and do not allow duplicate order_ids for the same merchant. | varchar(30) | Mandatory |
| Mode | This is the payment mode (TEST or LIVE are valid values) | varchar(4) | Optional |
| Amount | This is the payment amount. | decimal(12,2) | Mandatory |
| Currency | This is the 3-digit currency code (INR) | varchar(3) | mandatory |
| Description | Brief description of product or service that the customer is being charged for. | varchar(255) | mandatory |
| Name | Name of customer. | varchar(255) | mandatory |
| Email | Customer email address. | varchar(255) | mandatory |
| Phone | Customer phone number | varchar(30) | mandatory |
| address_line_1 | Customer address | varchar(255) | optional |
| address_line_2 | Customer address 2 | varchar(255) | optional |
| City | Customer city | varchar(255) | mandatory |
| State | Customer State | varchar(255) | optional |
| Country | Customer country | varchar(100) | mandatory |
| zip_code | Customer zip code | varchar(20) | mandatory |
| timeout_duration | Timeout duration (in seconds) | varchar(10) | optional |
| udf1 | User defined field | varchar(255) | optional |
| udf2 | User defined field 2 | varchar(255) | optional |
| udf3 | User defined field 3 | varchar(255) | optional |
| udf4 | User defined field 4 | varchar(255) | optional |
| udf5 | User defined field 5 | varchar(255) | optional |

| | | | |
|----------------------|--|----------------|-----------|
| return_url | Return URL success - Payfair Payments will make a POST request to this URL after successful transaction, with a set of parameters, which you can process as you want to. | varchar(255) | mandatory |
| return_url_cancel | Return URL success - Payfair Payments will make a POST request to this URL in case of transaction cancellation, with a set of parameters, which you can process as you want to. | varchar(255) | optional |
| percent_tdr_by_user | Percent of tdr amount paid by user (optional) (max value:100) | decimal(5,2) | optional |
| flatfee_tdr_by_user | fixed fee paid by user (optional) | decimal(10,2) | optional |
| show_convenience_fee | Controls whether the convenience fee amount (for surcharge merchants) is displayed to the customer (on the payment page) or not | varchar(1) | optional |
| split_enforce_strict | Controls whether payment is required to be split before settlement. By default, it is set to 'n', If this is set to 'y' then settlement will be on HOLD until splitsettlement api is called to provide split information. | varchar(1) | optional |
| split_info | Split info is for splitting the payment between vendor and themselves. In this field one must provide vendor code and what percentage of the payment to be split. (Note: Currently this accepts single vendor split amount percentage only) Following is an | varchar(500) | optional |

| | | | |
|---------------------------|---|--------------|----------|
| | <p>example how it will look</p> <pre>{"vendors": [{"vendor_code": "2VEN449", "split_amo unt_percentage": "20"}]}</pre> <p>All field in this JSON are mandatory.</p> | | |
| payment_options | <p>payment options to be displayed such as net banking (nb), wallet (w). Tabs will be displayed by order in which values are sent.</p> <p>Values accepted are: <i>NB,W,UPI</i> (comma separated string), sequence of values will also determine the tab sequence on payment page. 'NB': 'netbanking', 'W': 'wallet', 'UPI': 'upi',</p> | varchar(50) | optional |
| payment_page_display_text | This text will be displayed below the logo on payment page. | varchar(100) | optional |
| allowed_bank_codes | Bank codes sent in this filed will be allowed in payment page, other bank codes will not be allowed to proceed with payment. Refer appendix 3 for the list of bank codes. To send multiple bank codes send a comma separated list. E.g. to allow only credit cards: MACC,VICC,DINC,VISC,RUPC,MASC,AMXC | varchar(250) | optional |

| | | | |
|--------------------|---|--------------|----------|
| allowed_emi_tenure | This will be a comma separated integer list depending upon the tenure (in months) of loan allowed to show in EMI payment method. Ex(3 , 6, 9 etc.) | varchar(50) | optional |
| allowed_bins | BIN is Bank Identification Number, on a card it is first 6 digits. BINs passed here will only be allowed to transact, multiple BINs can be sent as comma separated list. Refer appendix 3 for the list if the payment mode is card | varchar(250) | optional |
| offer_code | If there is any discount / offer provided by merchant on EMIs, then predefined codes must be mentioned in this field. (This is for specific use case; more information can be provided on demand) | varchar(100) | optional |
| emi_info | This is an optional param which is to be posted to issuer end in case emi detail are required.the format will be as mentioned below <pre>"emi_info": { "subvention": "0.0", "aggregator_name": "BENOW", "bank_merchant_id": null, "bank_term_id": null, "bank_sku_code": null }</pre> all the fields are optional and have varchar as data type. | varchar(100) | optional |

| | | | |
|--------------------|---|---------------|-----------|
| product_details | <p>Contains information regarding the goods/product for which the payment (emi) is being made.</p> <p>Values in this field should be sent in JSON format, for example:</p> <pre>{"manufacturer": "Samsung", "category": "Phone", "sub_category_1": "Smart Phone", "sub_category_2": "High-end", "model_name": "Samsung Galaxy S10 Pro"} Fields such as manufacturer, category, model_name are self-explanatory; sub_category_1 and sub_category_2 further describe the variants/types of that product.</pre> <p>All fields in this JSON are optional.</p> | varchar(2048) | optional |
| enable_auto_refund | Payment request is auto refunded in case of delay success depending upon the value present in the field is 'y' or 'n'. If this filed is not sent default set for your account at the time of setup will take effect. | varchar(1) | optional |
| hash | You need to compute a hash of all your parameters and pass that hash to Payfair Payments, for details about calculating hash | varchar(255) | mandatory |

| | | | |
|-----------|---|-------------|-----------|
| | <p>refer Appendix 2.</p> <p>Note: the SALT will be provided by Payfair Payments separately.</p> <p>NEVER PASS SALT IN A FORM, DO NOT STORE SALT IN ANDROID APP APK or IPHONE APP package</p> | | |
| bank_code | Bank code identifies the payment mode and channel. | varchar(20) | Mandatory |

2.3. Response Parameters returned

| Parameter name | Description |
|------------------|---|
| transaction_id | A unique ID that can be used to trace the transaction uniquely within Payfair Payments. Transaction IDs are alphanumeric. An example transaction ID is HDVISC1299876438 |
| payment_mode | This tells the payment mode used by customer - example: "UPI", "netbanking", etc. |
| payment_channel | This tells the payment channel used by customer - example: "Visa", "HDFC Bank", "Paytm", etc. |
| payment_datetime | Date and Time of this payment in "DD-MM-YYYY HH:MM:SS" format |
| response_code | Status of the transaction (return code). 0 signifies successful transaction. Non-zero values signify error. Response Code returned is documented in Appendix 4 |
| response_message | The response message associated with the transaction. |
| error_desc | The detailed error description, if any |
| order_id | The same order_id that was originally posted by the merchant in the request. |
| amount | The same original amount that was sent by the merchant in the transaction request. In case of customer surcharge model this will be the amount paid by customer **. |
| currency | This is the 3digit currency code (INR), it will be same value that was originally sent by merchant. |

| | |
|------------------------|--|
| description | The same description that was originally sent by the merchant in the transaction request. |
| name | The same value that was originally sent by merchant |
| email | The same value that was originally sent by merchant |
| phone | The same value that was originally sent by merchant |
| address_line_1 | The same value that was originally sent by merchant |
| address_line_2 | The same value that was originally sent by merchant |
| city | The same value that was originally sent by merchant |
| state | The same value that was originally sent by merchant |
| country | The same value that was originally sent by merchant |
| zip_code | The same value that was originally sent by merchant |
| udf1 | The same value that was originally sent by merchant |
| udf2 | The same value that was originally sent by merchant |
| udf3 | The same value that was originally sent by merchant |
| udf4 | The same value that was originally sent by merchant |
| udf5 | The same value that was originally sent by merchant |
| tdr_amount | This is the TDR charged on the transaction ** |
| tax_on_tdr_amount | This is the Tax (GST) charged on the TDR Amount ** |
| amount_orig | This is the amount requested by merchant **. Typically, this will be same as the amount field, but in case of customer surcharge model this will be a different value. |
| cardmasked | Masked card number which was used to make the transaction **. For example, 437748*****0069 |
| 'emi_tenure' | If "send_emi_details" merchant param is enabled and emi_info request is received. |
| 'emi_rate_of_interest' | If "send_emi_details" merchant param is enabled and emi_info request is received. |
| hash | Payfair Payments calculates the hash using the same algorithm which was outlined earlier. Hence, the merchant needs to check whether this returned hash matches the calculated hash. |

** Note: This parameter will be returned as part of the response **only** if the merchant's account has been enabled for the same. Please speak to your Payfair Payments relationship manager if you would like this information to be returned in response.

Note: It is important to validate the hash after you receive the response from Payfair Payments. A failed response sent from Payfair Payments server to your server via browser could be tampered by malicious.

3. PAYMENT STATUS API

Payfair Payments provides an API which you can use to check the status of any prior transaction. You can use this to reconcile transactions. We strongly recommend that you make it a practice to use this for every transaction that was made. This serves two purposes:

The response might not reach you due to network issues or other problems such as user clicking refresh button on their browser, etc.

This also protects against any tampering, since you have a second fallback check here.

Payfair Payments offers a sophisticated API wherein you can apply "filters" on the resultset you want to retrieve. You can search our system by the transaction ID, or the order ID, or even by parameters such as date range, customer phone number, etc. You can also pass in various combinations of these parameters to get the resultset of your choice.

Note: Your designated server IP will need to be whitelisted by Payfair Payments for this API to work. If you receive errors such as "Unauthorized" while accessing this API, please contact your Payfair Payments relationship manager to get this fixed.

URL: <https://pay.payfair.in/v2/paymentstatus>

3.1. Parameters to be POSTed (JSON REQUEST)

| Parameter Name | Description | Data type | Optional / Mandatory |
|----------------|--|-------------|----------------------|
| api_key | Payfair Payments would assign a unique 36-digit merchant key to you. This key is exclusive to your business/login account. If you have multiple login accounts, there will necessarily be one different api_key per login account that is assigned to you. | varchar(36) | Mandatory |
| order_id | This is your (merchant) reference number which you submitted while making the original transaction. You can send multiple order ids in this field as comma (,) separated list | varchar(30) | Optional |
| transaction_id | This is the transaction ID generated by Payfair Payments for the given transaction | varchar(30) | Optional |

| | | | |
|----------------|---|--------------|-----------|
| bank_code | This is the 4-letter bankcode which denotes the payment mode/channel of the payment. | varchar(4) | Optional |
| response_code | The numeric response code returned by Payfair Payments during the original transaction | number(4) | Optional |
| customer_phone | Phone number of the customer, as provided during the original paymentrequest API | varchar(30) | Optional |
| customer_email | Email address of the customer, as provided during the original paymentrequest API | varchar(255) | Optional |
| customer_name | Name of the customer, as provided during the original paymentrequest API | varchar(255) | Optional |
| date_from | Start date of date range to retrieve transactions, in DD-MM-YYYY or YYYY-MM- DD HH:MM:SS format | varchar(20) | Optional |
| date_to | End date of date range to retrieve transactions, in DD-MM-YYYY or YYYY-MM- DD HH:MM:SS format | varchar(20) | Optional |
| page_number | Page number you need to retrieve, its value is limited by information received in the first response that is received | integer | Optional |
| per_page | Number of records need to see per page, this value should be between 1 and 50 | integer | Optional |
| hash | <p>You need to compute a hash of all your parameters and pass that hash to Payfair Payments, for details about calculating hash refer Appendix 2.</p> <p>Note: the SALT will be provided by Payfair Payments separately. NEVER PASS SALT IN A FORM, DO NOT STORE SALT IN ANDROID APP APK or IPHONE APP package</p> | varchar(255) | Mandatory |

3.2. Response Parameters

On successful call to this API you will receive JSON response. You can read the JSON response and process it at your end. If your result set is greater than 50 transactions, you would need to use pagination.

Note: few parameters in response will be visible only if it is enabled for your account, for example: refund_details will be available if it is enabled for your merchant account.

A few sample responses for given requests are provided

below: In case of success,

```
{  
  "data": [  
    {  
      "transaction_id": "SFSBIN2783912661",  
      "bank_code": "SBIN",  
      "payment_mode":  
        "Netbanking",  
      "payment_channel": "State Bank of  
        India", "payment_datetime": "2018-06-  
        13 16:44:03",  
      "response_code": 1000,  
      "response_message":  
        "FAILED",  
      "authorization_staus": null,  
      "order_id": "427641",  
      "amount": "27.36",  
      "amount_orig": "2.00",  
      "tdr_amount": 21.49,  
      "tax_on_tdr_amount": 3.87,  
      "description": "Web Payment for  
        433487", "error_desc": "FAILED",  
      "customer_phone": "9900990099",  
      "customer_name": "sharathkumar  
        hegde",  
    }]
```

```
"customer_email":  
"sharathkumar@example.com" "currency":  
"INR",  
"cardmasked": null,  
"udf1": null,  
"udf2": null,  
"udf3": null,  
"udf4": null,  
"udf5": null,  
"refund_details":  
": {  
    "refund_amount": 0  
}  
,  
{  
    "transaction_id":  
"HDVISC4291974106",  
    "bank_code": "VISC",  
    "payment_mode": "Credit Card",  
    "payment_channel": "Visa",  
    "payment_datetime": "2018-06-13  
16:45:39",  
    "response_code": 0,  
    "response_message":  
"SUCCESS",  
    "authorization_staus":  
"captured", "order_id":  
"427643",  
    "amount": "1.93",  
    "amount_orig": "1.90",  
    "tdr_amount": 0.03,  
    "tax_on_tdr_amount": 0,  
    "description": "Web Payment for  
433489", "error_desc": null,  
    "customer_phone": "9900990099",  
    "customer_name": "sharathkumar hegde",  
    "customer_email":  
"sharathkumar@example.com" "currency":  
"INR",
```

```

    "cardmasked": null,
    "udf1": null,
    "udf2": null,
    "udf3": null,
    "udf4": null,
    "udf5": null,
    "refund_details": {
        ": {
            "refund_amount": 0
        }
    },
    "page": {
        "total": 175,
        "per_page": 10,
        "current_page": 1,
        "last_page": 18,
        "from": 1,
        "to": 10
    },
    "hash": "30FAAD865191B4064576F063177F0A4692C3DBBBF35D1A20463EAA449269C4715FD13528EA069B3A8 D5C25C62637ED825C297C2337CDC1CFB7FCD0D60DCFEB9D"
}

```

In case of error,

```
{
    "error": {
        "code": 1001,
        "message": "The api key field is incorrect"
    }
}
```

In case there is no record present in our system for the combination of input, following error is returned

```
{
    "error": {
        "code": 1050,
```

```

    "message": "No data record found for the given input"
}
}

```

In case there is no transaction id in our system for the order_id, merchant_order_id or transaction_id, following error is returned

```

{
  "error": {
    "code": 1028,
    "message": "No Transaction found"
  }
}

```

If there are more than 50 transactions for which the status is requested, you would see following error

```

{
  "error": {
    "code": 1086,
    "message": "More than 50 records, refine your search criteria or use pagination"
  }
}

```

In such cases where result set is expected to have status of more than 50 transactions, it is required to use pagination.

Pagination allows to access data in smaller chunks making it easier for server to return data quickly. To allow for pagination one need to pass following additional parameter in request

| Parameter Name | Description | Data type | Optional / Mandator y |
|----------------|---|-----------|-----------------------|
| page_number | Page number you need to retrieve, its value is limited by information received in the first response that is received | integer | optional |
| per_page | Number of records need to see per page, this value should be between 1 and 50 | integer | optional |

If above parameters are passed, response will have additional information about pagination as following

```

"page": {

```

```
"total": 175,  
"per_page": 10,  
"current_page": 1,  
"last_page": 18,  
"from": 1,  
"to": 10  
},
```

This pagination information should be used in page_number filed for subsequent api request. end-user and turned into "success". To make sure the transaction response is the same as what Payfair Payments server sent please check the hash before considering the transaction response as final.

Note: Format of transaction ID is as follows: HDVISC1299876438". The 3rd to 6th digits (both inclusive) in the transaction ID signify the "bankcode". This information is enough to obtain the payment method and payment channel. A list of bankcodes and corresponding payment mode/channel is available in Appendix 3 of this document.

4. Refunds API

Payfair Payments provides a refund API which merchants can use to programmatically issue refunds instead of clicking the "refund" button in the Payfair Payments UI. This API can be invoked on any prior successful transaction. The transaction which is being refunded should be in either "paid" or "settled" state, or in "refunded" state (in case of partial amount refunds). Refunds can be either for the full amount paid by the customer, or any part of it.

The API needs a valid transaction ID as input.

Note: processing of refunds is subject to availability of funds in subsequent settlement cycles. This API will return a failure response in case sufficient funds are not available to process the refund.

4.1. Refund request API

URL: <https://pay.payfair.in/v2/refundrequest>

Request Parameters:

| Parameter Name | Description | Data type | Optional / Mandatory |
|----------------|---|-------------|----------------------|
| api_key | Payfair Payments would assign a unique 36- digit merchant key to you. This key is exclusive to your business/login account. If you have multiple login accounts, there will necessarily be one different api_key per login account that is assigned to you. | varchar(36) | Mandatory y |
| transaction_id | The unique alphanumeric transaction ID generated by Payfair Payments for a prior transaction. | varchar(30) | Mandatory y |

| | | | |
|--------------------|--|---------------|-----------|
| merchant_refund_id | This is your (merchant) refund reference number. It must be unique for every refund request. If a refund request is sent with same merchant_refund_id we return the response of the previously successful refund request. Warning: If you are NOT using this field then be careful, as each request will be treated as a new refund request. Thus it is recommended to use this field. | varchar(30) | Optional |
| merchant_order_id | This is your (merchant) reference number which you submitted while making the original transaction. Note that if this value does not match with related transaction_id field then you will get error. In typical cases do not send this field. | varchar(30) | Optional |
| amount | The amount which needs to be refunded. This needs to be less than or equal to the transaction amount. | decimal(10,2) | Mandatory |
| description | Description of the refund. Usually the reason for issuing refund, as specified by merchant. | varchar(500) | Mandatory |
| hash | You need to compute a hash of all your parameters and pass that hash to PayfairPayments, for details about calculating hash refer Appendix 2. Note: the SALT will be provided by Payfair Payments separately. NEVER PASS SALT IN A FORM, DO NOT STORE SALT IN ANDROID APP APK or IPHONE APP package | varchar(255) | Mandatory |

Response Parameters:

The output is a JSON which contains the error(s), if any, in validation, or a simple success message which confirms that the refund request has been accepted and will be processed during subsequent settlement cycle.

If the request is successfully processed response you will get a “*data*” block, and in case of failure you will see “*error*” block, you will not get “*data*” key in case of error.

In case of success, **NOTE:** that *refund_reference_no* is returned by the bank and it can be null in case refunds are not initiate by bank immediately, but is done at end of the day.

```
{  
  "data": {  
    "transaction_id":  
      "HDVISC7472820193", "refund_id":  
        4351, "refund_reference_no": null  
    "merchant_refund_id": 76783_R_1,  
    "merchant_order_id": 76783,  
  }  
}
```

In case of error,

```
{  
  "error": {  
    "code": 1039,  
    "message": "The refund amount is greater than transaction amount"  
  }  
}
```

4.2. Refund Status API

If a refund is initiated either from merchant or payment gateway end and merchant wants to check its status (details such as if it is refunded or not how much amount was paid and how much is refunded will be posted in response). To check the status of any refund which was initiated merchant should post the API request.

URL: <https://pay.payfair.in/v2/refundstatus>

Request Parameters:

| Parameter Name | Description | Data type | Optional / Mandatory |
|-------------------|---|--------------|----------------------|
| api_key | Payfair Payments would assign a unique 36- digit merchant key to you. This key is exclusive to your business/login account. If you have multiple login accounts, there will necessarily be one different api_key per login account that is assigned to you. | varchar(36) | Mandatory |
| transaction_id | The unique alphanumeric transaction ID generated by Payfair Payments for a prior transaction. | varchar(30) | Mandatory |
| merchant_order_id | This is your (merchant) reference number which you submitted while making the original transaction. Note that if this value does not match with related transaction_id field then you will get error. In typical cases do not send this field. | varchar(30) | Optional |
| hash | You need to compute a hash of all your parameters and pass that hash to PayfairPayments, for details about calculating hash refer Appendix 2. | varchar(255) | Mandatory |

| | | | |
|--|--|--|--|
| | <p>Note: the SALT will be provided by Payfair Payments separately.</p> <p>NEVER PASS SALT IN A FORM, DO NOT STORE SALT IN ANDROID APP APK or IPHONE APP Package</p> | | |
|--|--|--|--|

Response Parameters:

The output is a JSON it will give all details about a refund if any initiated for this transaction, if not it will give json with error. Partial multiple refunds are also shown in refund_details

If the request is successfully processed response you will get a “*data*” block, and in case of failure you will see “*error*” block, you will not get “*data*” key in case of error.

NOTE: that *refund_reference_no* is returned by the bank and it can be null in case refunds are not initiate by bank immediately but is done at end of the day.

```
{
  "data": {
    "transaction_id": "FDAXIP9740656834",
    "merchant_order_id": "351177",
    "refund_amount": 2.04,
    "transaction_amount": "2.04",
    "refund_details": [
      {
        "refund_id": 3523,
        "refund_reference_no": "602201803257434370",
        "merchant_refund_id": null,
        "refund_amount": "2.04",
        "refund_status": "Customer Refunded",
        "date": "2018-02-01 11:19:49"
      }
    ]
  },
}
```

```
"hash":  
"20D8CB42D14C35AAEF06BB200C82E560DCC1D0C19EEFFBF07CBEEB3BD39AE746AFB30  
A5803D6375 27CE1A45AE367565E8AF5933809E3F597D7CDDDCDB3C28FE"  
}
```

In case of error,

```
{  
"error": {  
"code": 1050,  
"message": "No data record found for the given input"  
}  
}
```

HASH CALCULATION GUIDE

Following are the steps to calculate hash.

1. Create a | (pipe) delimited string called hash_data with first value as the salt.
2. Now sort the post fields based on their keys and create a | delimited string, for the fields with values.
3. Hash the hash_data string using “**SHA-512**” hashing algorithm and save the hash in secure_hash string
4. Convert the secure_hash string to upper case

LIST OF RESPONSE CODE

| Error Num eric Code | Error Code | Error Description | Error Display Text |
|------------------------------|--------------------------------------|---|---|
| 0 | SUCCESS | Transaction successful | Transaction successful |
| 1000 | FAILED | Transaction failed | Transaction failed |
| 1001 | INVALID- API-KEY | The api key field is incorrect | The api key field is incorrect |
| 1002 | INVALID- LIVE- MODE- ACCESS | The live mode access is not allowed | The live mode access is not allowed |
| 1003 | INVALID- ORDER- ID-FIELD | The order id field should to be unique | The order id field should to be unique |
| 1004 | ORDER- ID-FIELD- NOT- FOUND | The order id field is not found | The order id field is not found |
| 1005 | INVALID- AUTHENT- ICATION | Invalid authentication at bank | Invalid authentication at bank |
| 1006 | WAITING- BANK- RESPONSE | Waiting for the response from bank | Waiting for the response from bank |
| 1007 | INVALID- INPUT- REQUEST | Invalid input in the request message | Invalid input in the request message |
| 1008 | TRANSAC- TION- TAMPERE- D | Transaction tampered | Transaction failed due to malicious activity in your account. Please contact the support team |
| 1009 | DECLINE- D-BY- BANK | Bank Declined Transaction | Bank Declined Transaction |
| 1010 | INVALID- AMOUNT | Amount cannot be less than 1 | Amount cannot be less than 1 |
| 1011 | AUTHORI- ZATION- REFUSED | Authorization refused | Authorization refused |
| 1012 | INVALID- CARD | Invalid Card/Member Name data | Invalid Card/Member Name data |
| 1013 | INVALID- EXPIRY- DATE | Invalid expiry date | Invalid expiry date |
| 1014 | DENIED- BY-RISK | Transaction denied by risk | Transaction denied by risk |
| 1015 | INSUFFICI- ENT- FUND | Insufficient Fund in Customer Account | Insufficient Fund in Customer Account |
| 1016 | INVALID- AMOUNT- LIMIT | Total Amount limit set for the terminal for transactions has been crossed | Total Amount limit set for the terminal for transactions has been crossed |

| | | | |
|------|-----------------------------|---|---|
| 1017 | INVALID-TRANSACTION-LIMIT | Total transaction limit set for the terminal has been crossed | Total transaction limit set for the terminal has been crossed |
| 1018 | INVALID-DEBIT-AMOUNT-LIMIT | Maximum debit amount limit set for the terminal for a day has been crossed | Maximum debit amount limit set for the terminal for a day has been crossed |
| 1019 | INVALID-CREDIT-AMOUNT-LIMIT | Maximum credit amount limit set for the terminal for a day has been crossed | Maximum credit amount limit set for the terminal for a day has been crossed |
| 1020 | MAXIMUM-DEBIT-AMOUNT-CROSS | Maximum debit amount set for per card for rolling 24 hrs has been crossed | Maximum debit amount set for per card for rolling 24 hrs has been crossed |
| 1021 | MAXIMUM-CREDIT-AMOUNT-CROSS | Maximum credit amount set for per card for rolling 24 hrs has been crossed | Maximum credit amount set for per card for rolling 24 hrs has been crossed |
| 1022 | MAXIMUM-TRANSACTION-CROSS | Maximum transaction set for per card for rolling 24 hrs has been crossed | Maximum transaction set for per card for rolling 24 hrs has been crossed |
| 1023 | HASH-MISMATCH | Hash Mismatch | Hash Mismatch |
| 1024 | INVALID-PARAMS | Invalid parameters | Invalid parameters |
| 9999 | UNKNOWN-WN-ERROR | Unknown error occurred | Unknown error occurred |
| 1025 | INVALID-BANK-CODE | Invalid bank code | Invalid bank code |
| 1026 | INVALID-MERCHANT | Merchant is not active | Merchant is not active |
| 1027 | INVALID-TRANSACTION | Invalid transaction | Transaction is invalid |
| 1028 | TRANSACTION-NOT-FOUND | Transaction not found | Transaction not found |
| 1029 | TRANSACTION- | Transaction terminated | Transaction terminated |

| | | | |
|------|--------------------------------------|---|---|
| | TERMINATED | | |
| 1030 | TRANSACTION-INCOMPLETED | Transaction incomplete | Transaction incomplete |
| 1031 | AUTO-REFUNDED | Transaction auto refunded | Transaction auto refunded |
| 1032 | REFUNDED | Transaction refunded | Transaction refunded |
| 1033 | SINGLE-TRANSACTION-LOWER-LIMIT-CROSS | The amount provided is less than transaction lower limit | The amount provided is less than transaction lower limit |
| 1034 | SINGLE-TRANSACTION-UPPER-LIMIT-CROSS | The amount provided is more than transaction upper limit | The amount provided is more than transaction upper limit |
| 1035 | TRANSACTION-DAILY-LIMIT-CROSS | The daily transaction limit is exceeded for the merchant | The daily transaction limit is exceeded for the merchant |
| 1036 | TRANSACTION-MONTHLY-LIMIT-CROSS | The monthly transaction limit is exceeded for the merchant | The monthly transaction limit is exceeded for the merchant |
| 1037 | DAILY-TRANSACTION-NUMBER-CROSS | The daily transaction number is exceeded for the merchant | The daily transaction number is exceeded for the merchant |
| 1038 | MONTHLY-TRANSACTION-NUMBER-CROSS | The monthly transaction number is exceeded for the merchant | The monthly transaction number is exceeded for the merchant |
| 1039 | INVALID-REFUND-AMOUNT | The refund amount is greater than transaction amount | The refund amount is greater than transaction amount |
| 1040 | INVALID-CVV | Invalid Card Verification Code | Invalid Card Verification Code |

| | | | |
|------|-----------------------|---|---|
| | AUTO- | | |
| 1041 | REFUNDE | Transaction is auto refunded by TnP | Transaction is auto refunded by TnP |
| | D-TNP | | |
| | FAILED- | | |
| 1042 | NO-RESPONSE | Transaction failed as there was no response from bank | Transaction failed as there was no response from bank |
| | E | | |
| | TRANSAC- | | |
| 1043 | TION-CANCELLED | Transaction cancelled | Transaction cancelled |
| | ED | | |
| 1044 | UNAUTHORIZED | Unauthorized | Unauthorized |
| 1045 | FORBIDDEN | Forbidden Access | Forbidden Access |
| | TRANSAC- | | |
| 1046 | TION-ALREADY-CAPTURED | Transaction already captured | Transaction already captured |
| | D | | |
| 1047 | AUTHORIZED | Transaction authorized | Transaction authorized |
| 1048 | CAPTURED | Transaction captured | Transaction captured |
| 1049 | VOIDED | Transaction voided | Transaction voided |
| | NO-RECORD-FOUND | No data record found for the given input | No data record found for the given input |
| 1050 | | | |
| 1051 | ACQUIRE-R-ERROR | Error occurred at the bank end | Error occurred at the bank end |
| 1052 | INVALID-EMAIL | Invalid Email ID | Invalid Email ID |
| 1053 | INVALID-PHONE | Invalid phone number | Invalid phone number |
| | SEAMLESS-NOT-ALLOWED | | |
| 1055 | D | Seamless payment not allowed | Seamless payment not allowed |
| 1056 | SESSION-TIMEOUT | Session expired. Please go back and try again. | Session expired. Please go back and try again. |
| 1057 | INVALID-VPA | Virtual Payee Address is invalid | Virtual Payee Address is invalid |
| 1058 | 3DS-FAILED | 3D Secure authentication failed | 3D Secure authentication failed |
| | ACCOUNT-BLOCKED | | |
| 1058 | T-BLOCKED | Bank Account is blocked | Bank Account is blocked |

| | | | |
|------|--|--|--|
| | ACQUIRE | | |
| 1059 | R-MAX- TRANSAC TION- LIMIT | Transaction amount limit has exceeded | Transaction amount limit has exceeded |
| 1060 | ACQUIRE R-MIN- TRANSAC TION- LIMIT | The amount provide is less than minimum transaction amount allowed | The amount provide is less than minimum transaction amount allowed |
| 1061 | ACQUIRE R- VELOCITY -LIMIT | Transaction frequency limit has exceeded | Transaction frequency limit has exceeded |
| 1062 | CARD- EXPIRED | Card has expired | Card has expired |
| 1063 | HOTLISTE D | Stolen or Lost Card | Stolen or Lost Card |
| 1064 | INACTIVE -CARD | The card is inactive | The card is inactive |
| 1065 | INCORRE CT-PIN | Incorrect PIN | Incorrect PIN |
| 1066 | INSUFFICI ENT- BALANCE | Insufficient Balance | Insufficient Balance |
| 1068 | INVALID- BIN | Bin not found | Bin not found |
| 1069 | INVALID- CARDHOL DER- NAME | Invalid card holder name | Invalid card holder name |
| 1070 | INVALID- CARD- INFO | Invalid brand or bin range not enabled | Invalid brand or bin range not enabled |
| 1071 | INVALID- INPUT- DATA | Invalid input data | Invalid input data |
| 1072 | ISSUER- DECLINE | Payment decline by bank | Payment decline by bank |
| 1073 | ISSUER- DISALLO WED | Transaction not allowed by the issuer | Transaction not allowed by the issuer |
| 1074 | ISSUER- ERROR | Payment processing failed due to error at bank | Payment processing failed due to error at bank |
| 1075 | ISSUER- LIMIT- DECLINE | Exceeds withdrawal frequency or count limit | Exceeds withdrawal frequency or count limit |
| 1076 | MAX- RETRIES | Maximum number of PIN retries exceeded | Maximum number of PIN retries exceeded |

| | | | |
|------|------------------------|--|---|
| 1077 | OTP-LIMIT-EXCEEDED | OTP validation attempts limit exceeded | OTP validation attempts limit exceeded |
| 1078 | PIN-ATTEMPT-S-EXCEEDED | PIN attempts limit exceeded | PIN attempts limit exceeded |
| 1079 | RESTRICTED-CARD | Restricted card | Restricted card |
| 1080 | SUSPECTED-FRAUD | Suspected fraud | Suspected fraud |
| 1081 | TIMEOUT | Payment was not completed on time | Payment was not completed on time |
| 1082 | UPI-PIN-NOT-SET | UPI PIN is not set | UPI PIN is not set |
| 1083 | UPI-REQUEST-EXPIRED | UPI request expired | UPI request expired |
| 1084 | USER-BACK-REFRESH | User pressed refresh button | User pressed refresh button |
| 1085 | ACQUIRE-R-DOWN | Payment acquirer is down | Payment acquirer is down |
| 1086 | TOO-MANY-RECORDS | Response contains too many records, use pagination | Too many records |
| 1087 | NOT-ENOUGH-PARAMETERS | Not enough parameters to get response, add one or more optional parameters | Not enough parameters, try adding one or more optional parameters |
| 1088 | TRANSACTION-IN-PROCESS | We are processing your transaction | We are processing your transaction |
| 1089 | INVALID-ACCESS-TOKEN | Invalid access token | Invalid access token |
| 1090 | TEST-MODE-DISABLED | The test mode access is not allowed | The test mode access is not allowed |
| 1091 | INVALID-TRANSFER-TYPE | The disbursement transfer type is not allowed | The disbursement transfer type is not allowed |
| 1092 | INACTIVE-ACCOUNT | The customers account is inactive | The customers account is inactive |

| | | | |
|------|--|--|--|
| 1093 | INVALID- ACCOUN T | The customers account is invalid | The customers account is invalid |
| 1094 | TRANSAC TION- COUNT- EXCEEDE D | Transaction count limit has exceeded for the customer | Transaction count limit has exceeded for the customer |
| 1095 | AMOUNT - MISMAT CH | Transaction tampered | Transaction amount mismatch in payment response |

LIST OF BANK CODES

| Bank Code | Payment Mode | Payment Channel | Description |
|------------------|---------------------|---|---|
| BHQR | Bharat QR | Bharat QR | Bharat QR Payments(Dynamic) |
| | Bharat | | |
| BSQR | QR(Static) | Bharat QR(Static) | Bharat QR Payments(Static) |
| EASF | Cardless EMI | EarlySalary | EarlySalary |
| INCF | Cardless EMI | Instacred | Instacred |
| ZEMF | Cardless EMI | ZestMoney | ZestMoney |
| DONH | Cash Card | DONE Cash Card | DONE Cash Card |
| ICSH | Cash Card | Icash Card | Icash card |
| ITZH | Cash Card | ITZ Cash Card | ITZ Cash card |
| PWMH | Cash Card | PayWorld Cash Card | PayWorld Cash Card |
| | Commercial | | |
| MACC | Credit Card | Master | Master Commercial Credit Card |
| | Commercial | | |
| VICC | Credit Card | Visa | Visa Commercial Credit Card |
| AMXC | Credit Card | Amex | American Express Credit Card |
| DINC | Credit Card | Diners | Diners Credit Card |
| MASC | Credit Card | Master | Master Credit Cards |
| RUPC | Credit Card | Rupay | Rupay Credit Card |
| VISC | Credit Card | Visa | Visa Credit Card |
| MAED | Debit Card | Maestro (non-SBI) | Maestro Debit Card (non-SBI) |
| MSED | Debit Card | Maestro (SBI) | Maestro Debit Card (SBI) |
| MASD | Debit Card | Master | Master Debit Card |
| RUPD | Debit Card | Rupay | Rupay Debit Card |
| VISD | Debit Card | Visa | Visa Debit Card |
| AXIL | Direct EMI | AXIS Bank Credit EMI | AXIS Bank Credit Card EMI |
| AXDL | Direct EMI | AXIS BANK DEBIT EMI | AXIS Bank Debit Card EMI |
| BOBL | Direct EMI | Bank of Baroda Credit EMI | Bank of Baroda Credit Card EMI |
| HDFL | Direct EMI | HDFC BANK Credit EMI | HDFC Bank Credit Card EMI |
| HDDL | Direct EMI | HDFC Bank Debit EMI | HDFC Bank Debit Card EMI |
| ICIL | Direct EMI | ICICI Bank Credit EMI | ICICI Bank Credit Card EMI |
| ICDL | Direct EMI | ICICI Bank Debit EMI | ICICI Bank Debit Card EMI |
| KKBL | Direct EMI | Kotak Mahindra Bank Credit EMI | Kotak Mahindra Bank Credit Card EMI |
| RBLL | Direct EMI | RBL Bank Credit Card EMI | RBL Bank Credit Card EMI |
| SCBL | Direct EMI | Standard Chartered Bank Credit Card EMI | Standard Chartered Bank Credit Card EMI |
| SBIL | Direct EMI | State Bank of India Credit EMI | State Bank of India Credit Card EMI |
| YESL | Direct EMI | Yes Bank Credit Card EMI | Yes Bank Credit Card EMI |
| EZP1 | EazyPay | EazyPay | EazyPay |
| ECLV | E-Collect | E-Collect | E-Collect(Virtual Accounts) |

| | | | |
|------|---------------------------|---|---|
| AMXE | EMI | American Express Credit EMI | American Express Credit EMI |
| AXIE | EMI | AXIS Bank Credit EMI | AXIS Bank Credit EMI |
| AXDE | EMI | AXIS BANK DEBIT EMI | AXIS Bank Debit EMI |
| BOBE | EMI | Bank of Baroda Credit EMI | Bank of Baroda Credit EMI |
| CITE | EMI | CITIBANK Credit EMI | CITIBANK Credit EMI |
| HDFE | EMI | HDFC Bank Credit EMI | HDFC Bank Credit EMI |
| HDDE | EMI | HDFC Bank Debit EMI | HDFC Bank Debit EMI |
| | | Hong Kong And Shanghai Banking Credit | Hong Kong And Shanghai Banking Credit |
| HSBE | EMI | EMI | EMI |
| ICIE | EMI | ICICI Bank Credit EMI | ICICI Bank Credit EMI |
| INDE | EMI | Indusind Bank Credit EMI | Indusind Bank Credit EMI |
| KKBE | EMI | Kotak Mahindra Bank Credit EMI | Kotak Mahindra Bank Credit EMI |
| RBLE | EMI | RBL Bank Credit EMI | RBL Bank Credit EMI |
| SCBE | EMI | Standard Chartered Bank Credit EMI | Standard Chartered Bank Credit EMI |
| SBIE | EMI | State Bank of India Credit EMI | State Bank of India Credit EMI |
| YESE | EMI | Yes Bank Credit EMI | Yes Bank Credit EMI |
| | International Credit Card | Amex | American Express International Credit Card |
| AMXI | Credit Card | Diners | Diners International Credit Card |
| | International Credit Card | Master | Master International Credit Card |
| VICI | Credit Card | Visa | Visa International Credit Card |
| | International Debit Card | Master | Master International Debit Card |
| MADI | Debit Card | Visa | Visa International Debit Card |
| ABPN | Netbanking | Aditya Birla Payments Bank | Aditya Birla Payments Bank |
| ADCN | Netbanking | Ahmedabad District Co-operative Bank | Ahmedabad District Co-operative Bank |
| AIRN | Netbanking | Airtel Payment Bank | Airtel Payment Bank |
| | | Allahabad Bank Corporate Netbanking | Allahabad Bank Corporate Netbanking |
| ALLM | Netbanking | (Erstwhile Indian Bank) | (Erstwhile Indian Bank) |
| | | Allahabad Bank NetBanking (Erstwhile Indian Bank) | Allahabad Bank NetBanking (Erstwhile Indian Bank) |
| ALLN | Netbanking | Indian Bank) | Indian Bank) |
| APGN | Netbanking | Andhra Pragathi Grameena Bank | Andhra Pragathi Grameena Bank |
| AUSN | Netbanking | AU Small Finance Bank | AU Small Finance Bank |
| AXIN | Netbanking | AXIS Bank | AXIS Bank NetBanking |
| AXIM | Netbanking | AXIS Bank - Corporate | AXIS Bank - Corporate NetBanking |
| BDNN | Netbanking | Bandhan bank | Bandhan Bank |
| BDNM | Netbanking | Bandhan bank - Corporate | Bandhan bank - Corporate |
| BBKN | Netbanking | Bank of Bahrain and Kuwait | Bank of Bahrain and Kuwait |
| BBRM | Netbanking | Bank of Baroda - Corporate | Bank of Baroda Corporate Banking |
| BBRN | Netbanking | Bank of Baroda - Retail | Bank of Baroda Retail Banking |
| BOIN | Netbanking | Bank of India | Bank of India |

| | | | |
|------|------------|---|---|
| BOMN | Netbanking | Bank of Maharashtra | Bank of Maharashtra |
| BRLM | Netbanking | Barclays Corporate Banking | Barclays Corporate Banking |
| BCBN | Netbanking | Bassein Catholic Bank | Bassein Catholic Bank |
| BCCN | Netbanking | Bassien Catholic Coop Bank | Bassien Catholic Coop Bank |
| BHAN | Netbanking | Bharat Bank | Bharat Bank |
| BMBN | Netbanking | Bharatiya Mahila Bank | Bharatiya Mahila Bank |
| CANN | Netbanking | Canara Bank | Canara Bank |
| CPFN | Netbanking | Capital Small Finance Bank | Capital Small Finance Bank |
| CSBN | Netbanking | Catholic Syrian Bank | Catholic Syrian Bank |
| CBIN | Netbanking | Central Bank Of India | Central Bank Of India |
| CITN | Netbanking | Citi Bank NetBanking | Citi Bank NetBanking |
| CUBN | Netbanking | City Union Bank | City Union Bank |
| COSN | Netbanking | Cosmos Bank | Cosmos Bank |
| DBSN | Netbanking | DBS Bank | DBS Bank |
| DCBM | Netbanking | DCB Bank - Corporate | DCB Bank - Corporate Netbanking |
| DEMN | Netbanking | Demo Bank | Demo Netbanking |
| DENN | Netbanking | Dena Bank | Dena Bank |
| DSHN | Netbanking | Deutsche Bank | Deutsche Bank |
| DCBN | Netbanking | Development Credit Bank | Development Credit Bank |
| DHNN | Netbanking | Dhanlakshmi Bank | Dhanalakshmi Bank |
| | | Dhanlakshmi Bank - Corporate Net | |
| DHNM | Netbanking | Banking | Dhanalakshmi Bank - corporate |
| ESFN | Netbanking | Equitas Bank | Equitas Small Finance Bank Limited |
| ESAN | Netbanking | ESAF Small Finance Bank | ESAF Small Finance Bank |
| FEDN | Netbanking | Federal Bank | Federal Bank |
| FESN | Netbanking | Federal Bank Scan And Pay | Federal Bank Scan And Pay |
| FSFN | Netbanking | Fincare Small Finance Bank | Fincare Small Finance Bank |
| GPPN | Netbanking | GP Parsik Sahakari Bank | GP Parsik Sahakari Bank |
| GSCN | Netbanking | Gujarat State Co-operative Bank Limited | Gujarat State Co-operative Bank Limited |
| HDFN | Netbanking | HDFC Bank | HDFC Bank |
| HSBN | Netbanking | HSBC NetBanking | HSBC NetBanking |
| ICIN | Netbanking | ICICI Bank | ICICI Netbanking |
| ICIM | Netbanking | ICICI Bank - Corporate | ICICI Bank - Corporate NetBanking |
| IDBN | Netbanking | IDBI | Industrial Development Bank of India |
| | | | Industrial Development Bank of India - |
| IDBM | Netbanking | IDBI Bank - Corporate | Corporate |
| IDFN | Netbanking | IDFC Bank | IDFC Bank |
| ININ | Netbanking | Indian Bank | Indian Bank |
| IOBN | Netbanking | Indian Overseas Bank | Indian Overseas Bank |
| INDN | Netbanking | IndusInd Bank | IndusInd Bank |
| INGN | Netbanking | ING Vysya Bank | ING Vysya Bank |
| JAKN | Netbanking | Jammu and Kashmir Bank | Jammu and Kashmir Bank |
| JSFN | Netbanking | Jana Small Finance | Jana Small Finance NetBanking |

| | | | |
|------|------------|---|---|
| JSBN | Netbanking | Janata Sahakari Bank | Janata Sahakari Bank |
| JPNB | Netbanking | Jio Payments Bank | Jio Payments Bank |
| KJBN | Netbanking | Kalyan Janata Sahakari Bank | Kalyan Janata Sahakari Bank |
| KRKN | Netbanking | Karnataka Bank | Karnataka Bank |
| KGBN | Netbanking | Karnataka Gramin Bank | Karnataka Gramin Bank |
| KVGN | Netbanking | Karnataka Vikas Grameena Bank | Karnataka Vikas Grameena Bank |
| KRVM | Netbanking | Karur Vysya - Corporate | Karur Vysya - Corporate Netbanking |
| KRVN | Netbanking | Karur Vysya - Retail | Karur Vysya |
| KKBN | Netbanking | Kotak Mahindra Bank | Kotak Mahindra Bank |
| LVBN | Netbanking | Laxmi Vilas Bank | Laxmi Vilas Bank - Retail |
| LVBM | Netbanking | Laxmi Vilas Bank - Corporate | Laxmi Vilas Bank - Corporate |
| MGBN | Netbanking | Maharashtra Gramin Bank | Maharashtra Gramin Bank |
| NBLN | Netbanking | Nainital Bank | Nainital Bank |
| NKBN | Netbanking | NKGSB Bank | NKGSB Bank |
| NEBN | Netbanking | North East Small Finance Bank | North East Small Finance Bank |
| NNBN | Netbanking | Nutan Nagarik Sahakari Bank Limited | Nutan Nagarik Sahakari Bank Limited |
| PYTN | Netbanking | Paytm Bank | Paytm Net Banking |
| OBCN | Netbanking | PNB (Erstwhile-Oriental Bank of Commerce) | PNB (Erstwhile-Oriental Bank of Commerce) |
| UNIN | Netbanking | PNB (Erstwhile-United Bank of India) | PNB (Erstwhile-United Bank of India) |
| PNYN | Netbanking | PNB YUVA Bank | PNB YUVA Bank |
| PMCN | Netbanking | Punjab & Maharashtra Coop Bank | Punjab & Maharashtra Coop Bank |
| PSBN | Netbanking | Punjab & Sind Bank | Punjab & Sind Bank |
| PNBM | Netbanking | Punjab National Bank - Corporate | Punjab National Bank-Corporate |
| PNBN | Netbanking | Punjab National Bank - Retail | Punjab National Bank - Retail Banking |
| RTNN | Netbanking | RBL Bank Limited | RBL Bank Limited |
| RTNM | Netbanking | RBL Bank Limited- Corporate Net Banking | RBL Bank Limited- Corporate Net Banking |
| RBSN | Netbanking | Royal Bank Of Scotland | Royal Bank Of Scotland NetBanking |
| SRSN | Netbanking | Saraswat Bank | Saraswat Bank |
| SOMN | Netbanking | SBM Bank India | SBM Bank India |
| SVCN | Netbanking | Shamrao Vitthal Co-operative Bank | Shamrao Vitthal Co-operative Bank |
| | | Shamrao Vitthal Co-operative Bank - | Shamrao Vitthal Co-operative Bank - |
| SVCN | Netbanking | Corporate | Corporate |
| SMCN | Netbanking | Shivalik Mercantile Cooperative Bank Ltd | Shivalik Mercantile Cooperative Bank Ltd |
| SOIN | Netbanking | South Indian Bank | South Indian Bank |
| SCBN | Netbanking | Standard Chartered Bank | Standard Chartered Bank |
| SBJN | Netbanking | State Bank of Bikaner and Jaipur | State Bank of Bikaner and Jaipur |
| SBHN | Netbanking | State Bank of Hyderabad | State Bank of Hyderabad |
| SBIN | Netbanking | State Bank of India | State Bank of India |
| SBMN | Netbanking | State Bank of Mysore | State Bank of Mysore |
| SBPN | Netbanking | State Bank of Patiala | State Bank of Patiala |
| SBTN | Netbanking | State Bank of Travancore | State Bank of Travancore |
| SUBN | Netbanking | Suryoday Small Finance Bank | Suryoday Small Finance Bank |

| | | | |
|------|--------------|--|--|
| SYDN | Netbanking | Syndicate Bank (Erstwhile Canara Bank) | Syndicate Bank (Erstwhile Canara Bank) |
| TMBN | Netbanking | Tamilnad Mercantile Bank Ltd. | Tamilnad Mercantile Bank Ltd. |
| TSCN | Netbanking | Tamilnadu State Coop Bank | Tamilnadu State Coop Bank |
| TBSN | Netbanking | Thane Bharat Sahakari Bank Ltd | Thane Bharat Sahakari Bank Ltd |
| | | The Kalupur Commercial Cooperative | The Kalupur Commercial Cooperative |
| KCCN | Netbanking | Bank Limited | Bank Limited |
| MSBN | Netbanking | The Mehsana Urban Co Op Bank Ltd | The Mehsana Urban Co Op Bank Ltd. |
| | | The Surat Peopleâ€™s Co-operative Bank | The Surat Peopleâ€™s Co-operative Bank |
| SPCN | Netbanking | Limited | Limited |
| SUCN | Netbanking | The Sutex Co-op Bank Ltd | The Sutex Co-op Bank Ltd |
| VCBN | Netbanking | The Varachha Co-operative Bank | The Varachha Co-operative Bank |
| TJSN | Netbanking | TJSB Bank | TJSB Bank |
| UCON | Netbanking | UCO Bank | UCO Bank |
| USFN | Netbanking | Ujjivan Small Finance Bank | Ujjivan Small Finance NetBanking |
| UBIM | Netbanking | Union Bank of India - Corporate | Union Bank - Corporate Netbanking |
| UBIN | Netbanking | Union Bank of India - Retail | Union Bank of India |
| | | Union Bank of India (Erstwhile Andhra | Union Bank of India (Erstwhile Andhra |
| ADBN | Netbanking | Bank) | Bank) |
| | | Union Bank of India (Erstwhile Andhra | Union Bank of India (Erstwhile Andhra |
| ADBM | Netbanking | Bank) - Corporate | Bank) - Corporate |
| | | Union Bank of India (Erstwhile | Union Bank of India (Erstwhile |
| CRPN | Netbanking | Corporation Bank) | Corporation Bank) |
| | | Union Bank of India (Erstwhile | Union Bank of India (Erstwhile |
| CRPM | Netbanking | Corporation Bank) - Corporate | Corporation Bank) - Corporate |
| UTSN | Netbanking | Utkarsh Small Finance Bank | Utkarsh Small Finance Bank |
| VRBN | Netbanking | Varachha Co-Operative Bank | Varachha Co-Operative Bank |
| VIJN | Netbanking | Vijaya Bank | Vijaya Bank |
| YESN | Netbanking | Yes Bank | Yes Bank |
| YESM | Netbanking | Yes Bank - Corporate | Yes Bank - Corporate |
| ZOBN | Netbanking | Zoroastrian Co-operative Bank | Zoroastrian Co-operative Bank |
| ICPL | PayLater | ICICI Pay Later | ICICI Pay Later |
| LZPY | PayLater | LazyPay | LazyPay |
| MBZP | PayLater | ZIP | ZIP |
| | | Pharmarack | |
| PYMI | Credit Card | Pharmarack | Pharmarack Credit Card |
| MASP | Prepaid Card | Master | Master Prepaid Card |
| RUPP | Prepaid Card | Rupay | Rupay Prepaid Card |
| VISP | Prepaid Card | Visa | Visa Prepaid Card |
| UPII | UPI | Unified Payments Intent Interface | Unified Payments Intent Interface |
| UPIU | UPI | Unified Payments Interface | Unified Payments Interface |
| ATLW | Wallet | Airtel Money | Airtel Money |
| AMRW | Wallet | Airtel Money Rwanda | Airtel Money Rwanda |
| AMPW | Wallet | Amazon Pay | Amazon Pay |
| BKSW | Wallet | bKash Wallet | bKash Wallet |
| CITW | Wallet | Citibank Reward Points | Citibank Reward Points |

| | | | |
|------|--------|-------------------|---------------------------|
| DCBW | Wallet | DCB Cippy | DCB Cippy |
| EZCW | Wallet | ezeClick | ezeClick Wallet |
| FRCW | Wallet | FreeCharge | FreeCharge Wallet |
| IDMW | Wallet | Idea Money | Idea Money |
| JNCW | Wallet | Jana Cash | Jana Cash |
| MBKW | Wallet | Mobikwik | Mobikwik Wallet |
| VDFW | Wallet | mPesa Wallet | mPesa - Vodafone Wallet |
| MRPW | Wallet | mRupee | mRupee Wallet |
| MTIW | Wallet | MTN Côte d'Ivoire | MTN Côte d'Ivoire |
| MTRW | Wallet | MTN Rwanda | MTN Rwanda |
| OLAW | Wallet | Ola Money | Ola Money Wallet |
| ORGW | Wallet | Orange Money | Orange Money Wallet |
| OXIW | Wallet | Oxigen | Oxigen Wallet |
| PYCW | Wallet | Paycash | Paycash wallet |
| PTMW | Wallet | Paytm | Paytm wallet |
| PZAW | Wallet | Payzapp | Payzapp wallet |
| PHPW | Wallet | PhonePe | PhonePe wallet |
| PNBW | Wallet | PNB Wallet | PNB Wallet |
| JIOW | Wallet | Reliance JioMoney | Reliance Jio Money wallet |
| SBIW | Wallet | SBI Buddy | SBI Buddy |
| TMW | | | |
| W | Wallet | The Mobile Wallet | The Mobile Wallet |
| YESW | Wallet | Yes Pay | Yes Pay |
| YPAW | Wallet | YpayCash | YpayCash |
| ZIPW | Wallet | ZipCash | ZipCash |