Quick PHP Payment Integration

BIJLIPAY PHP Payment Api Integration KIT

2020

This kit is for integrating Bijlipay Payment request API in your PHP application. This is intended to developers how need **quick PHP integration**. For a more detailed documentation see “Bijlipay Api Integration Document v1.3” (Request Bijlipay if you need detailed api documentation).

# Content

The Zip file contains one file “paymentrequest.php”.

# Requirement

You will need following for your api integration to work.

1. API Access: You need to be enabled by Bijlipay to use api integration.
2. API Key: You will need an api\_key, this will be provided by Bijlipay
3. Salt: This is like a password for you api access. This will be provided by Bijlipay.

# Usage

Follow the steps below:

1. Place the paymentrequest.php in suitable location on your server.
2. Edit this file and update your api\_key and salt.
3. Now submit a post request to this file from your application. Below is list of fields that are allowed to post.

## Fields needed to post to paymentrequest.php:

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Description | Data type | 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. | Unsigned  Integer | Yes |
| mode | This is the payment mode (TEST or LIVE are valid values) | Varchar(4) | Optional |
| amount | This is the payment amount. | Decimal(15,2) | Yes |
| currency | This is the 3digit currency code (**This value should be INR**) | Varchar(3) | Yes |
| description | Brief description of product or service that the customer is being charged for. | Varchar(500) | Yes |
| name | Name of customer. | Varchar(100) | Yes |
| email | Customer email address. | Varchar(100) | Yes |
| phone | Customer phone number | Varchar(50) | Yes |
| address\_line\_1 | Customer address line 1 | Varchar(100) | Optional |
| address\_line\_2 | Customer address line 2 | Varchar(100) | Optional |
| city | Customer city | Varchar(50) | Yes |
| state | Customer State | Varchar(50) | Optional |
| country | Customer country (**This value should be IND**) | Varchar(50) | Yes |
| zip\_code | Customer zip code | Varchar(20) | Yes |
| return\_url | Return URL – Bijlipay will make a POST request to this URL after transaction is completed (success or failure), with a set of parameters, which you can process as you want to. | Varchar(300) | Yes |
| udf1 | User defined field 1 | Varchar(300) | Optional |
| udf2 | User defined field 2 | Varchar(300) | Optional |
| udf3 | User defined field 3 | Varchar(300) | Optional |
| udf4 | User defined field 4 | Varchar(300) | Optional |
| udf5 | User defined field 5 | Varchar(300) | Optional |
|  |  |  |  |

## Fields returned to return\_url after success or failure response.

|  |  |
| --- | --- |
| Parameter name | Description |
| transaction\_id | A unique ID that can be used to trace the transaction uniquely within Bijlipay. Transaction IDs are alphanumeric. |
| payment\_method | This tells the payment method used by customer – example: “credit card”, “debit card”, “netbanking”, etc. |
| payment\_datetime | Date and Time of this payment in “YYYY-MM-DD HH:MM:SS” format \* |
| response\_code | Status of the transaction (return code). 0 signifies successful transaction. Non-zero values signify error. |
| response\_message | Can have a value of “success” or “failure”. |
| 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. |
| 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 |
| hash | Bijlipay calculates the hash using the same algorithm which was outlined earlier. Hence, the merchant needs to check whether this returned hash matches the hash that was originally passed. This is to make sure the response is not modified and you are seeing a legitimate response for current transaction. (If you need to know the hashing algorithm look at function hashCalculate($salt, $input) in paymentrequest.php) |
| error\_desc | Failure reason (if transaction is failed) |