



---

## **Zaakpay Integration Document**

---

Version 2.0

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Sign-Up</b>	<b>1</b>
<b>3</b>	<b>Get Merchant ID and Secret Key</b>	<b>3</b>
<b>4</b>	<b>Staging Credentials</b>	<b>6</b>
<b>5</b>	<b>Placing files on your server</b>	<b>7</b>
<b>6</b>	<b>Non Seamless Integration</b>	<b>8</b>
6.1	Checksum Calculation . . . . .	8
6.2	Request Parameters . . . . .	9
6.3	Response Parameters . . . . .	12
<b>7</b>	<b>Seamless Integration</b>	<b>13</b>
7.1	Checksum Calculation . . . . .	13
7.2	Request Parameters . . . . .	15
7.3	Response Parameters . . . . .	18
<b>8</b>	<b>Check API</b>	<b>20</b>
8.1	Request Parameters . . . . .	20
8.2	Response Parameters . . . . .	21
<b>9</b>	<b>Update API</b>	<b>22</b>
9.1	Request Parameters . . . . .	22
9.2	Response Parameters . . . . .	24
<b>10</b>	<b>Bank-Codes</b>	<b>25</b>
<b>11</b>	<b>Zaakpay API Responses</b>	<b>27</b>
11.1	Transact API Responses . . . . .	27
11.2	Check API Responses . . . . .	30
11.3	Update API Responses . . . . .	32
<b>12</b>	<b>Testing</b>	<b>32</b>

## List of Figures

1	Sign-Up . . . . .	1
2	Dashboard-Home . . . . .	2
3	Dashbaord-Developer Section . . . . .	3
4	Dashboard-URL Section . . . . .	4
5	Dashboard-Transaction Limits . . . . .	4
6	Dashboard-UI Section . . . . .	5

## List of Tables

1	Web-Redirect Request . . . . .	9
2	Web-Redirect Response . . . . .	12
3	S2S API Request . . . . .	15
4	S2S API Response . . . . .	18
5	Check API Request . . . . .	20
6	Check API Response . . . . .	21
7	Update API Request . . . . .	22
8	Update API Response . . . . .	24
9	Bank-Codes . . . . .	25
10	Transact-API Responses Codes . . . . .	27
11	Transact-API Response Codes(Wallet) . . . . .	29
12	Check-API Response Codes . . . . .	30
13	Update-API Response Codes . . . . .	32

## 1 Introduction

Zaakpay is an online payments platform that offers multiple payment methods to both an individual user and a business.

So, whether you are an ecommerce giant, a small spunky start-up or an individual user simply wanting to make payments to businesses, we have products that cater to all your needs.

This document describes the steps for technical integration process between merchant website/app and Zaakpay Payment Gateway for enabling online transactions. This document is covered in two sections. Section 1 covers website integration and Section 2 covers the APIs provided to the merchants.

## 2 Sign-Up

Signup for a business account on Zaakpay. After signing up and verifying your account follow the steps below:

- Login to Zaakpay on <https://www.zaakpay.com>

Figure 1: Sign-Up

Sign Up

Sign Up To Get Free Merchant Account.

test@user.com

.....

.....

9876543210

☒ I agree to the [Terms & Conditions](#) of Zaakpay

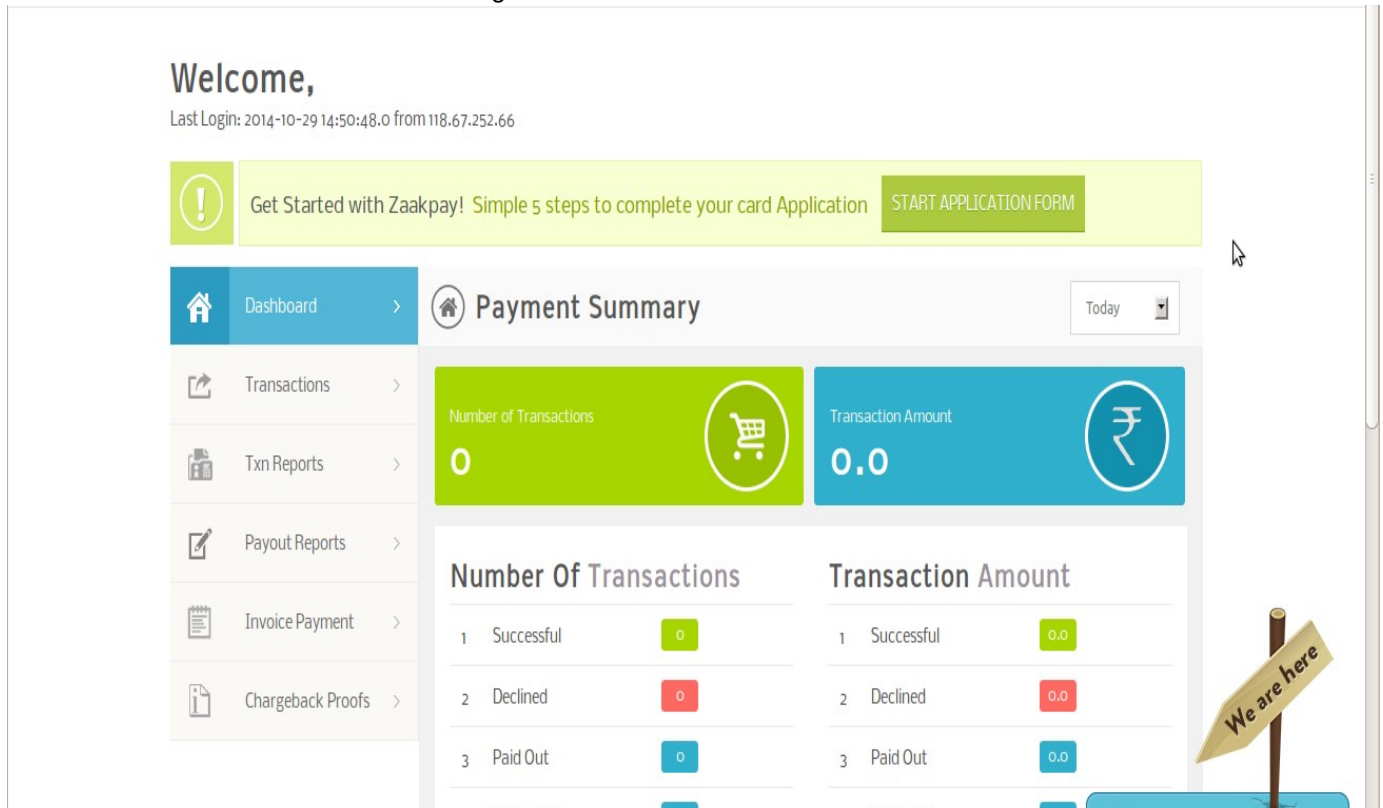
SUBMIT

Already have a Zaakpay merchant account? [Login](#)

- Click the My Account tab.
- Select the integration sub-menu item under the My Account tab.
- Select the URLs & Keys tab from the navigation.

- Fill in details like the domain you'll be posting from and your return URL. Here the domain is the domain where you'll be posting data to Zaakpay from and the response URL for transact API is the path to the response.ext file on your server.
- Select the Transaction limits sub-menu item under the My Account tab and set your appropriate transaction limits.

Figure 2: Dashboard-Home

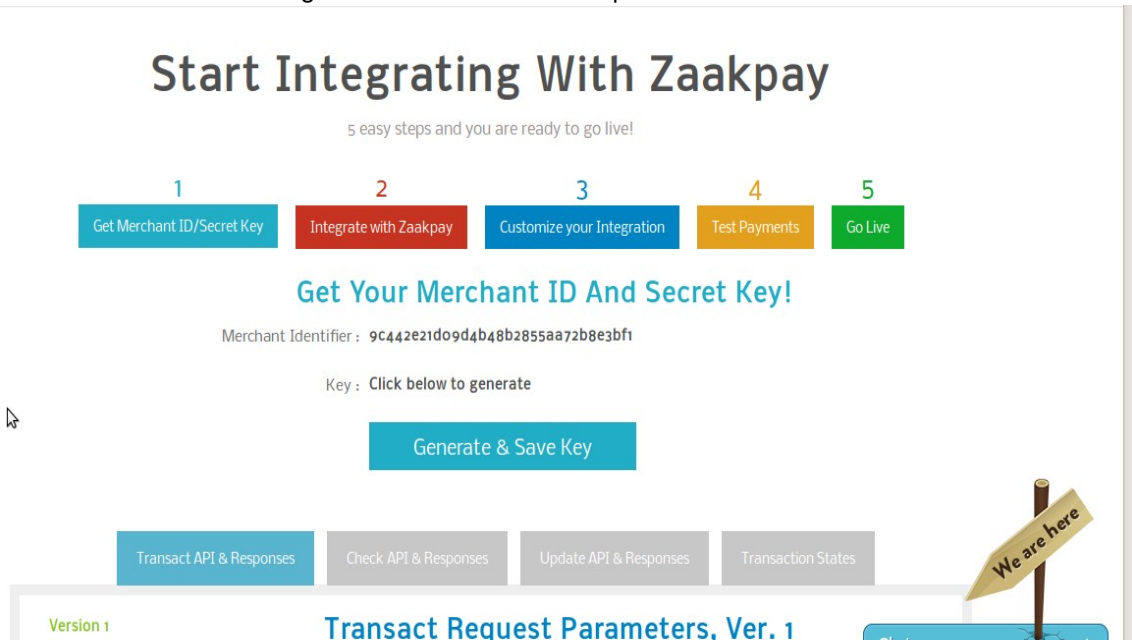


Generate your secret key and note it down along with your merchant identification number.

### 3 Get Merchant ID and Secret Key

Login to your Zaakpay account with registered email id. Go to Integration section. You'll get your Merchant Identifier and Secret key in URLs and Keys section.

Figure 3: Dashbaord-Developer Section



If Secret key is blank, you can generate Key by pressing the button "Generate Key" and save. If you're using the integration kit, please replace the values of the secret key in the response.ext and posttozaakpay.ext files where ext=extension.

Next, you need to fill in the domain details in your Zaakpay account. For that, click on "Customize your Integration" and then, click on "URL's" as described in the screen below.

Figure 4: Dashboard-URL Section

Domain Name :  
(Please include http:// or https://)

Transaction API return URL :  
(should be on the same domain)

Do you want to enable AUTO-CAPTURE for all your Zaakpay payments? Yes

Do you want to enable RETRY PAYMENT for all your Zaakpay payments? Yes

Do you want Zaakpay to send transaction emails to your customers? Yes

Do you want to receive transaction emails from Zaakpay? Yes

If you are agree with the above question, provide Email ID to get Transaction Emails

Save

After this, proceed to the next tab, "Transaction Limits". Here you can update the transaction caps (upper and lower) as per your requirements.

Figure 5: Dashboard-Transaction Limits  
**Customize Your Zaakpay Integration!**

URLs Transaction Limits UI on Zaakpay

Set your per transaction limits here

Max. Amount Per Transaction(In Rs.) 2000

Min. Amount Per Transaction(In Rs.) 100

Daily Max Number of Transactions Per User: 20

Daily Max Number of Transaction Per User Per Card : 20

Daily Max Number of Transaction Per User per Ip : 20

Save

Next you can complete the integration UI by uploading a brand image on the ext tab.

Figure 6: Dashboard-UI Section

### Customize Your Zaakpay Integration!

URLs

Transaction Limits

UI on Zaakpay

Customize your payment page on zaakpay.

Upload Logo (max-height:100px,max-width:230px):

Browse...

No file selected.

Your Logo:

Update Logo

Your Company or Brand name:

Save

Click on "Save" once you are done with all these configurations.

This was the overall set of procedures required for Zaakpay integration at our end. Next comes Merchant's side of integration, which is explained in the later sections.



## 4 Staging Credentials

- **URL** : `http://zaakpay-staging.cloudapp.net:8080`
- **Merchant Identifier** : `b19e8f103bce406cbd3476431b6b7973`
- **Secret key** : `0678056d96914a8583fb518caf42828a`
- **Public KeyId** : `sAMtcgidueVcrZl`
- **Public key** : `MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAikG2PaW+CqT3m26Dbtm7una22MYEDd+xONYjwE69Qa/FNQO0R5eqUnfi4lneWX6rc1IB6iVhyNDYULOZBW7vUsFbDWNJFDTD+V1T+30VXYvo+m7ufZCgxJVLn8W+JnKn1JPAL0n78UV2cG9zPIXKzJcMIGrNSG9QWFD6XJlriJ2CFEbzPf7a4y7DwNgGrRpqMkmJDHNLcaba+CtTqjgeGUWoVllg7RaQk4rJ5v21qyVK0pAUyfEXBDcLGWjsae0lsK+En7RFpV5NV6HxO78RnfT07RldIBHxjWeM9WJ+xuGBKrODXmKRdWXSCAliDYCp6F6fkgViE1XnCL6gQbnqQIDAQAB`

## 5 Placing files on your server

Based on your platform copy the posttozaakpay.ext, response.ext & checksum.ext files on your server. Here .ext depends on the programming language your server is based on.

For example, for PHP the file names will be response.php, etc.

- Make sure the code at the top of the files posttozaakpay.ext and response.ext is pointing to correct path you've placed checksum.ext (In case of checksum.jar you need to place it where other jar files libraries are located).

Example code :

```
// Hello.php
<%@ page import = "com.zaakpay.lib."ChecksumCalculator %>
<?php include'(checksum.'php); ?>
```

- Go to Zaakpay - My Account - Integration - URLs & Keys and update your transact API Return URL to the path where response.ext file is on your website.  
Example: <http://yourbrandname.com/payment/response.jsp>
- In the response.ext file subject to the checksum validation process, you may use/store the response data as you see fit. A copy of this data will be available in the Transaction History section of the Zaakpay Merchant Dashboard.

## 6 Non Seamless Integration

The purpose of this API is to enable websites to do online payment transactions using Zaakpay. Mobile Integration: For Zaakpay integration on mobile, parameter showMobile must be set to true. Everything else is same for desktop and mobile integration.

### 6.1 Checksum Calculation

For both integrity & data-authenticity verification before sending data to the API, you need to calculate a checksum of all the data that you send to Zaakpay. We use an HMACSHA- 256algorithm to calculate the checksum of ALL data that is posted to the API. We require data to be posted to our server in NVP (Name-Value Pairs) format.

To calculate the checksum please follow the process below:

- Create a list of all parameters which you're passing to the API. Parameters used in checksum calculation are in the same order as the order of posting the parameters to Zaakpay.
- Create a concatenated string of all data values in your list, with single quotes around each item. e.g. 'merchantIdentifier'orderId'amount'buyerEmail'buyerAddress'...
- Calculate the checksum using the HMAC SHA-256 algorithm using the concatenated string as data and your generated secret key.
- The resulting checksum calculated should be posted to the Zaakpay API along with other data. For example: Let's suppose we need to post the following data to the API. We calculate "checksum" only with the parameters mentioned below and the order of the parameters must remain intact when calculating "checksum".
  - merchantIdentifier -b19e8f103bce406cbd
  - orderId –223453
  - mode -1
  - merchantIpAddress –10.45.46.127
  - txnDate –2014-09-22

Now, we have to create a concatenated string of all the values, in the order in which they'll be sent to the API, with single quotes around each item. The string therefore will be:

'b19e8f103bce406cbd'223453'1'INR'20000'10.45.46.127'2013-05-23'

Now you can calculate the checksum based on this concatenated string and the secret key generated in your account under the URLs & Keys tab.

For more on HMAC implementations for various platforms please take a look at the following links:

- [PHP HMAC implementation](#)
- [Python HMAC implementation](#)
- [Perl HMAC implementation](#)
- [Ruby HMAC implementation](#)
- [C HMAC implementation](#)
- [Java implementation](#)
- [JavaScript HMAC implementation](#)

- [.NET's System.Security.Cryptography.HMAC](#)

The links provided above are for referential purposes only. The final checksum should be converted into HEXADECEIMAL character set.

Transact URL: <https://api.zaakpay.com/transact?v=3>

## 6.2 Request Parameters

Table 1: Web-Redirect Request

Parameter	Optional O, Mandatory M	Validation	Allowed Values
merchantIdentifier	M	alphanumeric	Zaakpay's unique identifier for your website
orderId	M	max 20 alphanumeric,must be unique per website, we do not accept duplicate	Your unique transaction identifier.
returnUrl	O	This must be the domain(or a sub-domain of it) you saved under My Account>Integration	Url where you want Zaakpay to post the response
buyerEmail	M	valid email address of the buyer	eg. prasang.misra@mobikwik.com
buyerFirstName	M	Max 30 alphanumeric characters, no special characters or dashes. First Name on card	Prasang
buyerLastName	M	Max 30 alphanumeric characters, no special characters or dashes. First name and last name cannot be same. Last Name on card.	Misra
buyerAddress	M	100 alphanumeric Street address of the buyer. (Part of billing address)	B-34, Priyadarshni Society, Dumna Road
buyerCity	M	30 alphabet, minimum 3 (Part of billing address)	Jabalpur
buyerState	M	State of the buyer (Part of billing address)	MP
buyerCountry	M	Country of the buyer	India
buyerPincode	M	Buyer's pin/zip code. Can have Numbers, Spaces and Hyphens (-) only ( Part of billing address )	482001
buyerPhoneNumber	M	buyer's landline or mobile phone number, numeric only, no dashes,no spaces	eg. 7698189874
txnType	M	1 digit only, numeric Zaakpay will show the tab on the payment page which corresponds to the txnType you provide	debit/credit cards=1, net banking=3
zpPayOption	M	Which Zaakpay option have you used for this transaction. 1 digit only, numeric Default value is 1.	0=on_zaakpay, 1=button_redirect, 2=widget_redirect, 3=api
mode	M	1 digit only, numeric	1 = Domain check, 0=Domain check skip

currency	M	Values defined by Zaakpay	INR
amount	M	Value in paisa. Min 100 paisa Max 10000000. Amount limit saved under Transaction Limit in your Zaakpay panel.	
merchantIpAddress	M	buyer's IP address as recorded by your website.	127.0.0.1
txnDate	M	Transaction date in yyyy-mm-dd format	2016-08-20
purpose	M	Min and max numeric 1 digit. You must specify the purpose of the transaction	0=Service, 1=Goods, 2=Auction, 3=Other
productDescription	M	Text description of what you are selling. Atleast 1 product description is mandatory to show in the bill on payment page. free text alphanumeric 100 max	e.g. name of book, name of mobile etc. e.g. Rs 199 Godzilla Movie DVD
product1Description	O	free text alphanumeric 100 max	
product2Description	O	free text alphanumeric 100 max	
shipToAddress	O	You may specify this only when buyer's address is different from shipping address. 30 alphanumeric	Flat 1A, Sector7, Defence Colony
shipToCity	O	Shipping address city. 30alphabet, minimum3	Jabalpur
shipToState	O	Shipping address state	MP
shipToCountry	O	Shipping address country	India
shipToPincode	O	Shipping address pin/zip code. 2 to 12 digits Can have Numbers, Spaces and Hyphens (-)only	482001
shipToPhoneNumber	O	Shipping address landline or mobile phone number numeric only, no dashes,no spaces	e.g. 01145771775 ,9971712962
shipToFirstname	O	max 30 alphanumeric characters,no special characters or dashes	Prasang
shipToLastname	O	max 30 alphanumeric characters,no special characters or dashes	Misra
showMobile	O	false:We show the full-fledged version unconditionally. DETECT:We do detection of the user Agent of the browser from which the request is sent& route accordingly. true:We show the mobile page unconditionally. missing/not sent: Same as DETECT (i.e. We do detection at our end ).	Only allowed value is "true" if you want Zaakpay to represent mobile view.
checksum	M	To be calculated on above parameters using HMAC SHA 256	

Example: Since you are sending payment information to Zaakpay, you need to prefill form parameters as hidden fields as a part of a form. Here is an example of what a form sending information to Zaakpay looks like:

```
<form action="https://api.zaakpay.com/transact?v=3" method="post">
<input type="hidden" name="merchantIdentifier" value="
b19e8f103bce406cbd">
<input type="hidden" name="orderId" value="444221414">
<input type="hidden" name="returnUrl" value="www.domain.com/zaakpay/
response">
<input type="hidden" name="buyerEmail" value="example@gmail.com">
<input type="hidden" name="buyerFirstName" value="kumar">
<input type="hidden" name="buyerLastName" value="prasant">
<input type="hidden" name="buyerAddress" value="Isa">
<input type="hidden" name="buyerCity" value="noida">
<input type="hidden" name="buyerState" value="u.p.">
<input type="hidden" name="buyerCountry" value="India">
<input type="hidden" name="buyerPincode" value="201012">
<input type="hidden" name="buyerPhoneNumber" value="9871041425">
<input type="hidden" name="txnType" value="1">
<input type="hidden" name="zpPayOption" value="1">
<input type="hidden" name="mode" value="1">
<input type="hidden" name="currency" value="INR">
<input type="hidden" name="amount" value="200000">
<input type="hidden" name="merchantIpAddress" value="127.0.0.1">
<input type="hidden" name="purpose" value="1">
<input type="hidden" name="productDescription" value="test_product">
<input type="hidden" name="product1Description" value="">
<input type="hidden" name="product2Description" value="">
<input type="hidden" name="product3Description" value="">
<input type="hidden" name="product4Description" value="">
<input type="hidden" name="shipToAddress" value="">
<input type="hidden" name="shipToCity" value="">
<input type="hidden" name="shipToState" value="">
<input type="hidden" name="shipToCountry" value="">
<input type="hidden" name="shipToPincode" value="">
<input type="hidden" name="shipToPhoneNumber" value="">
<input type="hidden" name="shipToFirstname" value="">
<input type="hidden" name="shipToLastname" value="">
<input type="hidden" name="txnDate" value="2011-08-30">
<input type="hidden" name="checksum" value="796
d672eb63e1dfa4a0bhjh67hkh98"> </form>
```

The main files in test kit are described below:

- **test\_merchant\_input.htm:** This is the entry point which has all details like amount, orderId, merchantIdentifier etc. => Put your merchant identifier here.  
**Note:** orderId must be unique for every request.
- **test\_mtx\_update\_input.htm:** This is the entry point for update/refund status API call. Put your details and merchant Identifier in this file. Amount for partial refund should be '0' in case of full refund.
- **test\_status\_check\_input.htm:** This is the entry point for check status API call. Put your order ID and merchant Identifier here. ext refers to your respective technology extension like for PHP, replace

ext with php

- **posttozaakpay.ext**: It populates all parameters in hidden variables and makes a POST request to Zaakpay. => Put your secret key here.
- **postmtxupdatetozaakpay.ext**: It makes the server to server update/refund API call and displays the returned message.
- **poststatuschecktozaakpay.ext**: It makes the server to server check status API call and displays the returned message.
- **response.ext**: This is the file which handles response from Zaakpay after completion of transaction. It has the code to fetch response parameters and calculate response checksum. => Put your secret key here.
- **Checksum.ext**: Has the code for calculating request checksum.

### 6.3 Response Parameters

Table 2: Web-Redirect Response

Parameters	Description
orderId	Your unique transaction identifier
responseCode	Numeric, max 3 digits example 100 for success
responseDescription	Alphanumeric max 30 description of the response
amount	Txn amount in paisa, Integer
paymentMethod	Payment Method ID for Card and Net Banking transactions. For Card txns, payment Method ID starts with C and N for Net Banking. It is alphanumeric value with max length 6. First letter is C or N, followed by 5 digits max.
cardhashid	Unique id for each card number used in transaction. For Netbanking txns, value will be "NA".
checksum	Checksum calculated by Zaakpay on all above response parameters

- **paymentMethod**: This parameter helps in determining the mode of payment. This parameter returns a unique id which is mapped to different cards/banks. For example, if the value of this parameter is N1001, payment was made using HDCF NetBanking. If the value is C4300, payment was made using Axis VISA Debit Card. In case of Mobikwik Wallet, value of this parameter is N1053.
- **cardhashid**: This is a one to one mapping with a card number. It is a unique value generated per card and will remain same for all transactions made using same card. This can help a merchant to extract information like how many transactions and of how much worth were made using a card. Merchants can also setup some fraud checks and limits per card using this parameter. In case of NetBanking and Mobikwik Wallet, value of this parameter is NA.
- **checksum**: Similar to request checksum, response checksum must be calculated on all response parameters by merchant and matched with the checksum sent by Zaakpay in response. Sample code to calculate response checksum has been given in file test\_merchant\_output.jsp

## 7 Seamless Integration

The purpose of this API is to enable websites to do online payment transactions using Zaakpay. Mobile Integration: For Zaakpay integration on mobile, parameter showMobile must be set to true. Everything else is same for desktop and mobile integration.

There are two flows in this section:

- **PCI-DSS certified merchants:**

In this case, the Transact URL will be: **<https://api.zaakpay.com/transactD?v=5>**

Here the card details will be encrypted on the merchant's server using RSA encryption.

- **PCI-DSS non-certified merchants:**

In this case, the Transact URL will be: **<https://api.zaakpay.com/transactD?v=3>**

Here the card details will be encrypted on Zaakpay's server using a js file.

### 7.1 Checksum Calculation

For both integrity & data-authenticity verification before sending data to the API, you need to calculate a checksum of all the data that you send to Zaakpay. We use an HMACSHA-256 algorithm to calculate the checksum of ALL data that is posted to the API. We require data to be posted to our server in NVP (Name-Value Pairs) format.

To calculate the checksum please follow the process below:

- Create a list of all parameters which you're passing to the API. Parameters used in checksum calculation are (in particular order):
  - merchantIdentifier
  - orderId
  - mode
  - currency
  - amount
  - merchantIpAddress
  - txnDate
- Create a concatenated string of all data values in your list, with single quotes around each item. e.g. 'merchantIdentifier"orderId"mode"currency"amount"merchantIpAddress"txnDate'
- Calculate the checksum using the HMAC SHA-256 algorithm using the concatenated string as data and your generated secret key.
- The resulting checksum calculated should be posted to the Zaakpay API along with other data. For example: Let's suppose we need to post the following data to the API. We calculate "checksum" only with the parameters mentioned below and the order of the parameters must remain intact when calculating "checksum".
  - merchantIdentifier -b19e8f103bce406cbd
  - orderId -223453
  - mode -1
  - currency - INR
  - amount - 200
  - merchantIpAddress -10.45.46.127



– txnDate –2014-09-22

Now, we have to create a concatenated string of all the values, in the order in which they'll be sent to the API, with single quotes around each item. The string therefore will be:

'b19e8f103bce406cbd"223453"1"INR"200"10.45.46.127"2014-09-22'

Now you can calculate the checksum based on this concatenated string and the secret key generated in your account under the URLs & Keys tab.

For more on HMAC implementations for various platforms please take a look at the following links:

- [PHP HMAC implementation](#)
- [Python HMAC implementation](#)
- [Perl HMAC implementation](#)
- [Ruby HMAC implementation](#)
- [C HMAC implementation](#)
- [Java implementation](#)
- [JavaScript HMAC implementation](#)
- [.NET's System.Security.Cryptography.HMAC](#)

The links provided above are for referential purposes only. The final checksum should be converted into HEXADECIMAL character set.

## 7.2 Request Parameters

Table 3: S2S API Request

Parameter	Optional O, Mandatory M	Validation	Allowed Values
merchantIdentifier	M	alphanumeric	Zaakpay's unique identifier for your website
orderId	M	max 20 alphanumeric,must be unique per website, we do not accept duplicate	Your unique transaction identifier.
returnUrl	O	This must be the domain(or a sub-domain of it) you saved under My Account>Integration	Url where you want Zaakpay to post the response
buyerEmail	M	valid email address of the buyer	eg. prasang.misra@mobikwik.com
buyerFirstName	M	Max 30 alphanumeric characters, no special characters or dashes. First Name on card	Prasang
buyerLastName	M	Max 30 alphanumeric characters, no special characters or dashes. First name and last name cannot be same. Last Name on card.	Misra
buyerAddress	M	100 alphanumeric Street address of the buyer. (Part of billing address)	B-34, Priyadarshni Society, Dumna Road
buyerCity	M	30 alphabet, minimum 3 (Part of billing address)	Jabalpur
buyerState	M	State of the buyer (Part of billing address)	MP
buyerCountry	M	Country of the buyer	India
buyerPincode	M	Buyer's pin/zip code. Can have Numbers, Spaces and Hyphens (-) only ( Part of billing address )	482001
buyerPhoneNumber	M	buyer's landline or mobile phone number, numeric only, no dashes,no spaces	eg. 7698189874
txnType	M	1 digit only, numeric Zaakpay will show the tab on the payment page which corresponds to the txnType you provide	debit/credit cards=1, net banking=3
zpPayOption	M	Which Zaakpay option have you used for this transaction. 1 digit only, numeric Default value is 1.	0=on_zaakpay, 1=button_redirect, 2=widget_redirect, 3=api
mode	M	1 digit only, numeric	1 = Domain check, 0=Domain check skip
currency	M	Values defined by Zaakpay	INR
amount	M	Value in paisa. Min 100 paisa Max 10000000. Amount limit saved under Transaction Limit in your Zaakpay panel.	

merchantIpAddress	M	buyer's IP address as recorded by your website.	127.0.0.1
txnDate	M	Transaction date in yyyy-mm-dd format	2016-08-20
purpose	M	Min and max numeric 1 digit. You must specify the purpose of the transaction	0=Service, 1=Goods, 2=Auction, 3=Other
productDescription	M	Text description of what you are selling. Atleast 1 product description is mandatory to show in the bill on payment page. free text alphanumeric 100 max	e.g. name of book, name of mobile etc. e.g. Rs 199 Godzilla Movie DVD
product1Description	O	free text alphanumeric 100 max	
product2Description	O	free text alphanumeric 100 max	
shipToAddress	O	You may specify this only when buyer's address is different from shipping address. 30 alphanumeric	Flat 1A, Sector7, Defence Colony
shipToCity	O	Shipping address city. 30alphabet, minimum3	Jabalpur
shipToState	O	Shipping address state	MP
shipToCountry	O	Shipping address country	India
shipToPincode	O	Shipping address pin/zip code. 2 to 12 digits Can have Numbers, Spaces and Hyphens (-)only	482001
shipToPhoneNumber	O	Shipping address landline or mobile phone number numeric only, no dashes,no spaces	e.g. 01145771775 ,9971712962
shipToFirstname	O	max 30 alphanumeric characters,no special characters or dashes	Prasang
shipToLastname	O	max 30 alphanumeric characters,no special characters or dashes	Misra
showMobile	O	false:We show the full-fledged version unconditionally. DETECT:We do detection of the user Agent of the browser from which the request is sent& route accordingly. true:We show the mobile page unconditionally. missing/not sent: Same as DETECT (i.e. We do detection at our end ).	Only allowed value is "true" if you want Zaakpay to represent mobile view.
debitorcredit	M	Possible Values: debit, credit, net banking or wallet	
bankid	M (for Net Banking)	For Net Banking, ID of selected bank, as SBI	
encrypted_pan	M (for Card txn)	Encrypted Card Number	
card	O	Card type: VISA,MasterCard	
nameoncard	M (for Card txn)	Card Holder Name	

encryptedcvv	M (for Card txn)	Encrypted CVV of card	
encrypted_expiry_month	M (for Card txn)	Encrypted Expiry Month of card	
encrypted_expiry_year	M (for Card txn)	Encrypted Expiry year of card	
checksum	M	To be calculated on above parameters using HMAC SHA 256	

The card details need to be encrypted and sent across the https POST parameters. This encryption can be done by the help of RSA encryption. Example: Since you are sending payment information to Zaakpay, you need to pre-fill form parameters as hidden fields as a part of a form. Here is an example of what a form sending information to Zaakpay looks like:

```
<form action="https://api.zaakpay.com/transactD?v=3" method="post">
<input type="hidden" name="merchantIdentifier" value="
b19e8f103bce406cbd">
<input type="hidden" name="orderId" value="444221414">
<input type="hidden" name="returnUrl" value="">
<input type="hidden" name="buyerEmail" value="a@b.com">
<input type="hidden" name="buyerFirstName" value="Prasang">
<input type="hidden" name="buyerLastName" value="Misra">
<input type="hidden" name="buyerAddress" value="jbp">
<input type="hidden" name="buyerCity" value="Jabalpur">
<input type="hidden" name="buyerState" value="M.P.">
<input type="hidden" name="buyerCountry" value="India">
<input type="hidden" name="buyerPincode" value="482001">
<input type="hidden" name="buyerPhoneNumber" value="7698189874">
<input type="hidden" name="txnType" value="1">
<input type="hidden" name="zpPayOption" value="1">
<input type="hidden" name="mode" value="1">
<input type="hidden" name="currency" value="rupee">
<input type="hidden" name="amount" value="200000">
<input type="hidden" name="merchantIpAddress" value="127.0.0.1">
<input type="hidden" name="purpose" value="1">
<input type="hidden" name="productDescription" value="test_product">
<input type="hidden" name="product1Description" value="">
<input type="hidden" name="product2Description" value="">
<input type="hidden" name="product3Description" value="">
<input type="hidden" name="product4Description" value="">
<input type="hidden" name="shipToAddress" value="">
<input type="hidden" name="shipToCity" value="">
<input type="hidden" name="shipToState" value="">
<input type="hidden" name="shipToCountry" value="">
<input type="hidden" name="shipToPincode" value="">
<input type="hidden" name="shipToPhoneNumber" value="">
<input type="hidden" name="shipToFirstname" value="">
<input type="hidden" name="shipToLastname" value="">
<input type="hidden" name="txnDate" value="2011-08-30">
<input type="hidden" name="debitorcredit" value="wallet" />
<input type="hidden" name="encrypted_pan" value="" />
<input type="hidden" name="card" value="" />
```

```

<input type="hidden" name="nameoncard" value="" />
<input type="hidden" name="encryptedcvv" value="" />
<input type="hidden" name="encrypted_expiry_month" value="" />
<input type="hidden" name="encrypted_expiry_year" value="" />
<input type="hidden" name="checksum"
value="796
d672eb63e1dfa4a0bc844e8d3468ebcd6d612dc39588814b7b00ce669c1c2">
</form>

```

The main files in test kit are described below:

- **test\_merchant\_input.htm:** This is the entry point which has all details like amount, orderid, merchantIdentifier etc. => Put your merchant identifier here.  
**Note:** orderId must be unique for every request.
- **test\_mtx\_update\_input.htm:** This is the entry point for update/refund status API call. Put your details and merchant Identifier in this file. Amount for partial refund should be '0' in case of full refund.
- **test\_status\_check\_input.htm:** This is the entry point for check status API call. Put your order ID and merchant Identifier here. ext refers to your respective technology extension like for PHP, replace ext with php
- **posttozaakpay.ext:** It populates all parameters in hidden variables and makes a POST request to Zaakpay. => Put your secret key here.
- **postmtxupdatetozaakpay.ext:** It makes the server to server update/refund API call and displays the returned message.
- **poststatuschecktozaakpay.ext:** It makes the server to server check status API call and displays the returned message.
- **response.ext:** This is the file which handles response from Zaakpay after completion of transaction. It has the code to fetch response parameters and calculate response checksum. => Put your secret key here.
- **Checksum.ext:** Has the code for calculating request checksum.

### 7.3 Response Parameters

Table 4: S2S API Response

Parameters	Description
orderId	Your unique transaction identifier
responseCode	Numeric, max 3 digits example 100 for success
responseDescription	Alphanumeric max 30 description of the response
amount	Txn amount in paisa, Integer
paymentMethod	Payment Method ID for Card and Net Banking transactions. For Card txns, payment Method ID starts with C and N for Net Banking. It is alphanumeric value with max length 6. First letter is C or N, followed by 5 digits max.
cardhashid	Unique id for each card number used in transaction. For Netbanking txns, value will be "NA".
checksum	Checksum calculated by Zaakpay on all above response parameters

- **paymentMethod:** This parameter helps in determining the mode of payment. This parameter returns a unique id which is mapped to different cards/banks. For example, if the value of this parameter is N1001, payment was made using HDCF NetBanking. If the value is C4300, payment was made using Axis VISA Debit Card. In case of Mobikwik Wallet, value of this parameter is N1053.
- **cardhashid:** This is a one to one mapping with a card number. It is a unique value generated per card and will remain same for all transactions made using same card. This can help a merchant to extract information like how many transactions and of how much worth were made using a card. Merchants can also setup some fraud checks and limits per card using this parameter. In case of NetBanking and Mobikwik Wallet, value of this parameter is NA.
- **checksum:** Similar to request checksum, response checksum must be calculated on all response parameters by merchant and matched with the checksum sent by Zaakpay in response. Sample code to calculate response checksum has been given in file test\_merchant\_output.jsp

## 8 Check API

The purpose of this API is to enable websites to check the latest status of their transaction at any time.  
Check status URL: <https://api.zaakpay.com/checktransaction?v=2>

### 8.1 Request Parameters

Table 5: Check API Request

Parameter	Optional O, Mandatory M	Validation	Allowed Values
merchantIdentifier	M	alphanumeric	
orderId	M	Transaction id for which you want to check the status	Your unique transaction identifier
mode	M	1 digit only, numeric	0
checksum	M	Checksum calculated on all above request parameters	

The parameters must be posted to the Update Transaction API using HTTP(POST). Apart from the listed parameters, a checksum is also expected. Refer below section for clarification on checksum generation.

#### Checksum Calculation:

Create a list of all parameters which you're passing to the API. Parameters used in checksum calculation are (in no particular order):

- merchantIdentifier
- mode
- orderId

Create a concatenated string of all data values in your list, with single quotes around each item. Calculate the checksum using the HMAC SHA-256 algorithm using the concatenated string as data and your generated secret key.

The resulting checksum calculated should be posted to the Zaakpay API along with other data. For example: Let's suppose we need to post the following data to the API. We calculate "checksum" with the parameters mentioned below:

- merchantIdentifier -b19e8f103bce406cbd
- mode - 0
- orderId - ZPK12345

Now, we have to create a concatenated string of all the values, in the order in which they'll be sent to the API, with single quotes around each item. The string therefore will be:

**'b19e8f103bce406cbd'0'ZPK12345'**

Now you can calculate the checksum based on this concatenated string and the secret key generated in your account under the URLs & Keys tab.

Example:

```

<form action="https://api.zaakpay.com/checktransaction?v=2" method="
  post">
  <input type="hidden" name="merchantIdentifier" value="
    sk4jfdb342kjwdbkj9"/>
  <input type="hidden" name="orderId" value="897698973"/>
  <input type="hidden" name="mode" value="0"/>
  <input type="hidden" name="checksum" value="
    e2d9328528939568cc252e45aadb8"/>
</form>

```

## 8.2 Response Parameters

The response will be in the XML format.

Table 6: Check API Response

Parameters	Description
merchantid	Zaakpay's unique identifier for your website
orderid	Your unique transaction identifier
responsecode	Numeric, max 3 digits example 100 for success
description	Alphanumeric max 30 description of the response
paymentmethod	Payment Method ID for Card and Net Banking transactions. For Card txns, payment Method ID starts with C and N for Net Banking. It is alphanumeric value with max length 6. First letter is C or N, followed by 5 digits max.
cardhashid	Unique id for each card number used in transaction. For Netbanking txns, value will be "NA".
amount	Txn amount in paisa, Integer
checksum	Checksum calculated by Zaakpay on all above response parameters

Example:

```

<zaakpay_response>
<response_element>
<merchantid>sk4jfdb342kjwdbkj9</merchantid>
<orderid>99802312</orderid>
<responsecode>190</responsecode>
<description>OrderId either not Processed or Rejected</description>
<paymentmethod>NOT FOUND</paymentmethod>
<cardhashid>NA</cardhashid>
<amount>25000</amount>
<checksum>yun863a2d9328528939568cc252e45aadb8</checksum>
</response_element>
</zaakpay_response>

```



## 9 Update API

The purpose of this API is to enable websites to settle, cancel or refund transactions. Update API URL: <https://api.zaakpay.com/updatetransaction>

### 9.1 Request Parameters

Table 7: Update API Request

Parameter	Optional O, Mandatory M	Validation	Allowed Values
merchantIdentifier	M	alphanumeric	Zaakpay unique merchant identifier for your website
orderId	M	Max 20 alphanumeric, must be unique per website, we do not accept duplicate	Your unique transaction identifier
mode	M	1 digit only, numeric	0
updateDesired	M	Numeric max1digit, values predefined by Zaakpay	7="Captured", 8="Canceled", 14="Refunded", 22="Partial Refund". Note:If you request a state update to "Refunded"we will issue the full amount refund to the user.
updateReason	M	Description of the reason for update. min5, max 50 alphanumeric characters. no special characters or dashes	Examples: you want to cancel a transaction, your user wants a refund, you want to settle a transaction
amount	O(during Full-Refund),M(for Partial-Refund)	Amount in paisa. Amount which needs to be refunded in case of partial refunds. In case of full refund this can be omitted.	example Re1 is 100 paisa, Rs 777.50 is 77750 paisa. Pass this parameter if merchant wants partial refund.
checksum	M	Checksum calculated on all above request parameters	

The parameters may be posted to the Update Transaction API using HTTP(POST). Apart from the listed parameters, a checksum is also expected. Refer below section for clarification on checksum generation.

Create a list of all parameters which you're passing to the API.Parameters used in checksum calculation are(in no particular order):

- merchantIdentifier
- updateDesired
- updateReason
- orderId
- mode

Create a concatenated string of all data values in your list, with single quotes around each item. Calculate the checksum using the HMAC SHA-256 algorithm using the concatenated string as data and your generated secret key.

The resulting checksum calculated should be posted to the Zaakpay API along with other data.

For example: Let's suppose we need to post the following data to the API. We calculate "checksum" with the parameters mentioned below:

- merchantIdentifier - b19e8f103bce406cbd
- updateDesired - 7
- updateReason - Test Reason
- orderId - ZPK12345
- Mode - 0

Now, we have to create a concatenated string of all the values, in the order in which they'll be sent to the API, with single quotes around each item. The string therefore will be:

**'b19e8f103bce406cbd'7'Test Reason'ZPK12345'0'**

Now you can calculate the checksum based on this concatenated string and the secret key generated in your account under the URLs & Keys tab.

**Note:**

Only 3 kinds of updates are possible using Update API:

- Authorized to Cancel
- Authorized to Capture
- Capture to Refund before Payout Initiated
- Capture to Partial Refund before Payout Initiated
- Payout Initiated to Refund Initiated
- Payout Initiated to Partial Refund Initiated
- Payout Completed to Refund Initiated
- Payout Completed to Partial Refund Initiated

Example:

```
<form action="https://api.zaakpay.com/updatetransaction" method="
post">
<input type="hidden" name="merchantIdentifier" value="
sk4j2kjwdbkj9832ds" />
<input type="hidden" name="orderId" value="897698973" />
<input type="hidden" name="mode" value="1" /> ...
<input type="hidden" name="updateDesired" value="14" />
<input type="hidden" name="updateReason" value="Order_not_delivered.
" />
<input type="hidden" name="checksum" value="
"a2d9328528939568cc252e45aadb8" />
</form>
```

## 9.2 Response Parameters

The response will be in the XML format.

Table 8: Update API Response

Parameters	Description
merchantid	Zaakpay's unique identifier for your website
orderid	Your unique transaction identifier
responsecode	Numeric, max 3 digits example 100 for success
description	Alphanumeric max 30 description of the response
checksum	Checksum calculated by Zaakpay on all above response parameters

Example:

```
<zaakpay_response>
  <response_element>
    <merchantid>sk4j2kjwdbkj9832ds</merchantid>
    <orderid>99802312</orderid>
    <responsecode>190</responsecode>
    <description>OrderId either not Processed or Rejected</description>
  </response_element>
</zaakpay_response>
```

## 10 Bank-Codes

This category contains the codes for net-banking as well as the wallet services that we currently offer. Below is a combined list of both.

Table 9: Bank-Codes

Bank Code	Bank Name
HDF	HDFC Bank
ALB	Allahabad Bank
ADB	Andhra Bank
BBK	Bank of Bahrain and Kuwait
BBC	Bank of Baroda - Corporate Banking
BBR	Bank of Baroda - Retail Banking
BOI	Bank of India
BOM	Bank of Maharashtra
CNB	Canara Bank
CSB	Catholic Syrian Bank
CBI	Central Bank of India
CUB	City Union Bank
CRP	Corporation Bank
DEN	Dena Bank
DBK	Deutsche Bank
DCB	Development Credit Bank
DLB	Dhanalakshmi Bank
FBK	Federal Bank
IDB	IDBI Bank
INB	Indian Bank
IOB	Indian Overseas Bank
IDS	IndusInd Bank
ING	ING Vysya Bank
JKB	Jammu and Kashmir Bank
KBL	Karnataka Bank Ltd
KVB	Karur Vysya Bank
162	Kotak Bank
LVC	Laxmi Vilas Bank - Corporate Net Banking
LVR	Laxmi Vilas Bank - Retail Net Banking
OBC	Oriental Bank of Commerce
PSB	Punjab and Sind Bank
CPN	Punjab National Bank - Corporate Banking
PNB	Punjab National Bank - Retail Banking
RTN	Ratnakar Bank
SVC	Shamrao Vitthal Co-operative Bank
SIB	South Indian Bank
SBJ	State Bank of Bikaner and Jaipur
SBH	State Bank of Hyderabad
SBM	State Bank of Mysore
SBP	State Bank of Patiala
SBT	State Bank of Travancore
SYD	Syndicate Bank
TMB	Tamilnad Mercantile Bank Ltd.

UCO	UCO Bank
UBI	Union Bank of India
VJB	Vijaya Bank
YBK	Yes Bank Ltd
SBI	State Bank of India
ICICI	ICICI Bank
AXIS	Axis Bank
UNIZP	United Bank of India
MW	Mobikwik Wallet
EZE	Amex Eze Click
IDEBIT	ICICI ATM+Pin
HDFZP	HDFC Bank
MSPASS	Masterpass
icashw	ICASH CARD
PAYUWL	PayU Wallet
OXYW	Oxigen Wallet
payzpw	Hdfc Payzapp Wallet
IDN	IDFC Bank

## 11 Zaakpay API Responses

### 11.1 Transact API Responses

Table 10: Transact-API Responses Codes

Response Code	Response Description	Is Success
100	The transaction was completed successfully.	✓
101	Merchant not found. Please check your merchantIdentifier field.	✗
102	Customer cancelled transaction.	✗
103	Fraud Detected.	✗
104	Customer Not Found.	✗
105	Transaction details not matched.	✗
106	IpAddress BlackListed.	✗
107	Transaction Amount not in specified amount range.	✗
108	Validation Successful.	✗
109	Validation Failed.	✗
110	MerchantIdentifier field missing or blank.	✗
111	MerchantIdentifier Not Valid.	✗
126	Date received with request was not valid.	✗
127	ReturnUrl does not match the registered domain.	✗
128	Order Id Already Processed with this Merchant.	✗
129	OrderId field missing or blank.	✗
130	OrderId received with request was not Valid.	✗
131	ReturnUrl field missing or blank.	✗
132	ReturnUrl received with request was not Valid	✗
133	BuyerEmail field missing or blank.	✗
134	BuyerEmail received with request was not Valid.	✗
135	BuyerFirstName field missing or blank.	✗
136	BuyerFirstName received with request was not Valid.	✗
137	BuyerLastName field missing or blank	✗
138	BuyerLastName received with request was not Valid	✗
139	BuyerAddress field missing or blank.	✗
140	BuyerAddress received with request was not Valid.	✗
141	BuyerCity field missing or blank.	✗
142	BuyerCity received with request was not Valid.	✗
143	BuyerState field missing or blank	✗
144	BuyerState received with request was not Valid.	✗
145	BuyerCountry field missing or blank.	✗
146	BuyerCountry received with request was not Valid.	✗
147	BuyerPincode field missing or blank.	✗
148	BuyerPinCode received with request was not Valid.	✗
149	BuyerPhoneNumber field missing or blank.	✗
150	BuyerPhoneNumber received with request was not Valid.	✗
151	TxnType field missing or blank.	✗
152	TxnType received with request was not Valid.	✗
153	ZpPayOption field missing or blank.	✗
154	ZpPayOption received with request was not Valid.	✗
155	Mode field missing or blank	✗
156	Mode received with request was not Valid.	✗

157	Currency field missing or blank.	X
158	Currency received with request was not Valid.	X
159	Amount field missing or blank.	X
160	Amount received with request was not Valid.	X
161	BuyerIpAddress field missing or blank	X
162	BuyerIpAddress received with request was not Valid.	X
163	Purpose field missing or blank.	X
164	Purpose received with request was not Valid.	X
165	ProductDescription field missing or blank.	X
166	ProductDescription received with request was not Valid.	X
167	Product1Description received with request was not Valid.	X
168	Product2Description received with request was not Valid.	X
169	Product3Description received with request was not Valid.	X
170	Product4Description received with request was not Valid.	X
171	ShipToAddress received with request was not Valid.	X
172	ShipToCity received with request was not Valid.	X
173	ShipToState received with request was not Valid.	X
174	ShipToCountry received with request was not Valid.	X
175	ShipToPincode received with request was not Valid.	X
176	ShipToPhoneNumber received with request was not Valid.	X
177	ShipToFirstname received with request was not Valid	X
178	ShipToLastname received with request was not Valid.	X
179	Date is blank.	X
179	Date received with request was not valid.	X
180	Checksum received with request is not equal to what we calculated.	X
181	Merchant Data Complete.	X
182	Merchant data not completed in our database	X
183	Unfortunately, the transaction has failed	X
400	The transaction was declined by the issuing bank	X
401	The transaction was rejected by the acquiring bank	X
402	This test transaction has been successfully completed.	X
403	Transaction failed because this card has been blocked by Zaaipay	X
404	Transaction failed due to security checks	X
501	Debitorcredit is blank	X
502	Bankid is blank	X
503	Encrypted pan is blank	X
504	Card is blank	X
505	Nameoncard is blank	X
506	Encrypted cvv is blank	X
507	Encrypted expiry month is blank	X

The below response code series starting from '6' e.g. '6XX' are sent from MobiKwik wallet via Zaakpay to merchant site.

Table 11: Transact-API Response Codes(Wallet)

Response Code	Response Description	Is Success
601	Transaction completed successfully	✓
602	Merchant secret key doesn't exist	✗
603	User blocked	✗
604	Merchant blocked	✗
605	Merchant doesn't exist	✗
606	Merchant not registered on MobiKwik	✗
607	Wallet Topup failed	✗
608	Wallet debit failed	✗
609	Wallet credit failed	✗
610	User canceled transaction at login page	✗
611	User cancelled transaction at Wallet Top Up page	✗
612	User cancelled transaction at Wallet Debit page	✗
613	Order Id already processed with this merchant	✗
614	Length of parameter orderid must be between 8 to 30 characters	✗
615	Parameter orderid must be alphanumeric only	✗
616	Parameter email is invalid	✗
618	Parameter cell is invalid. It must be numeric, have 10 digits and start with 7,8,9	✗
619	Parameter merchantname is invalid. It must be alphanumeric and its length must be between 1 to 30 characters	✗
620	Parameter redirecturl is invalid	✗
621	User Authentication failed	✗
622	Monthly Wallet Top up limit crossed	✗
623	Monthly transaction limit for this user crossed	✗
624	Maximum amount per transaction limit for this merchant crossed	✗
625	Merchant is not allowed to perform transactions on himself	✗
626	Checksum Mismatch	✗
627	Unexpected Error	✗
628	Orderid is Blank or Null	✗
629	Unknown Error	✗



## 11.2 Check API Responses

Table 12: Check-API Response Codes

Response Code	Response Description	Transaction Success	Valid for refund
103	Fraud Detected	✗	✗
110	MerchantIdentifier field missing or blank	✗	✗
111	MerchantIdentifier not valid	✗	✗
129	OrderId field missing or blank	✗	✗
155	Mode field missing or blank	✗	✗
156	Mode received with request was not valid	✗	✗
180	Checksum received with request is not equal to what we calculated.	✗	✗
182	Merchant Data not complete in our database.	✗	✗
89	Checksum was blank.	✗	✗
190	OrderId either not processed or Rejected.	✗	✗
191	Merchant Identifier or Order Id was not valid.	✗	✗
205	We could not find this transaction in our database.	✗	✗
206	Transaction in Scheduled state.	✗	✗
207	Transaction in Initiated state.	✗	✗
208	Transaction in Processing state.	✗	✗
209	Transaction has been authorized.	✗	✗
210	Transaction has been put on hold.	✗	✗
211	Transaction is incomplete.	✗	✗
212	Transaction has been settled.	✓	✗
213	Transaction has been canceled.	✗	✗
223	Data Validation success.	✗	✗
228	Transaction has been captured.	✓	✓
230	Transaction Refund Initiated	✓	✗
231	Transaction Refund Completed	✓	✗
232	Transaction Payout Initiated	✓	✓
233	Transaction Payout Completed	✓	✓
234	Transaction Payout Error.	✗	✗
236	Transaction Refund Paid Out	✓	✗
237	Transaction Chargeback has been initiated	✓	✗
238	Transaction Chargeback is being processed	✓	✗
239	Transaction Chargeback has been accepted	✓	✗
240	Transaction Chargeback has been reverted	✓	✗
241	Transaction Chargeback revert is now complete	✓	✗
245	Transaction Partial Refund Initiated	✓	✓
246	Transaction Partial Chargeback has been initiated	✓	✓
247	Transaction Partial Chargeback is being processed	✓	✓
248	Transaction Partial Chargeback has been accepted	✓	✓
249	Transaction Partial Chargeback has been reverted	✓	✓
251	Transaction Partial Refund Paid out	✓	✓

252	Transaction Partial Refund Completed	✓	✓
253	Transaction Refund Before Payout Paid out	✓	✓
254	Transaction Partial Refund Before Payout Paid Out	✓	✓
255	Transaction Partial Refund Before Payout Completed	✓	✓
256	Transaction Refund Before Payout Completed	✓	✗
400	Your Bank has declined this transaction, please Retry this payment with another Card.	✗	✗

### 11.3 Update API Responses

Table 13: Update-API Response Codes

Response Code	Response Description	Update Success
184	Update Desired blank.	✗
185	Update Desired not Valid	✗
186	Update Reason blank.	✗
187	Update Reason Not Valid.	✗
189	Checksum was blank.	✗
190	orderId either not Processed or Rejected.	✗
201	Transaction cannot be refunded.	✗
203	Transaction status could not be updated try again.	✗
229	Transaction cannot be captured.	✗
230	Transaction Refund Initiated	✓
242	Transaction captured successfully.	✓
243	Transaction canceled successfully.	✓
245	Transaction Partial Refund Initiated	✓

## 12 Testing

If everything works as it should, after a payment is completed you should be directed back to your website along with POST data about the result & other parameters of the transaction. This part is handled by the response.ext file, which displays all the received information and also verifies the checksum to verify the integrity of the information received. The parameters received with a response from the Zaakpay transact API can be seen on this page.

You should take the response.ext as a starting point and accordingly display the end result to your customers and other things.

For Example: In case of a successful responseCode & successful checksum verification you can display a success page to the customer and show his order has been placed successfully. You can also keep a copy of the transaction details in your database by updating it for each response received here.