

Onebip Web Payments

Integration Guide 3.3

Notice of non-liability

Onebip is providing the information in this document to you AS-IS with all faults. Onebip makes no warranties of any kind (whether express, implied or statutory) with respect to the information contained herein. Onebip assumes no liability for damages (whether direct or indirect), caused by errors or omissions, or resulting from the use of this document or the information contained in this document or resulting from the application or use of the product or service described herein. Onebip reserves the right to make changes to any information herein without further notice. Onebip does not guarantee that the features described in this document will be announced or made available to anyone in the future.

Index

THIS DOCUMENT	4
INTENDED AUDIENCE	4
1. INTRODUCTION	5
2. INTEGRATION TYPES	5
2.1 STANDARD.....	5
2.2 EXPRESS	7
3 ONE-TIME PAYMENTS	8
3.1 CONSTRUCTING ONE-TIME PAYMENT REQUESTS	8
3.2 PROCESSING THE ONEBIP TRANSACTION RESPONSE	12
3.2.1 NOTIFICATION PARAMETERS.....	13
3.2.2 MD5 ENCRYPTION OF THE NOTIFICATION URL.....	16
3.3 RETURN URL	17
4 SUBSCRIPTIONS	18
4.1 CONSTRUCTING SUBSCRIPTION REQUESTS	18
4.2 PROCESSING THE ONEBIP TRANSACTION RESPONSE	19
4.2.1 NOTIFICATION PARAMETERS.....	20
4.2.2 MD5 ENCRYPTION OF THE NOTIFICATION URL.....	20
4.3 RETURN URL	22
5 API OPERATIONS	23
1.1 DOWNLOAD REPORTS.....	23
1.2 CANCEL SUBSCRIPTION.....	24
6 APPENDIX	25
6.1 NATION CODES	25
6.2 CURRENCY CODES	26
6.3 LANGUAGE CODES.....	27
6.4 SUBSCRIPTION-BASED PRICE POINTS (VAT INCL.).....	28
6.5 ONEBIP “ADD FUNDS” PRICE POINTS (VAT INCL.)	29
6.6 EXPRESS PRICE POINTS (VAT INCL.)	30

This Document

This document describes how to receive Onebip payments from your website with a customised integration.

Even though Onebip can be integrated in its simplest form just with basic web development skills, you will also need familiarity with scripting languages if you use a dynamic platform to handle customer registrations, logins, and transactions, and you want Onebip to be integrated with it.

Intended Audience

This document is written for merchants and developers who want to configure and test their Onebip-based applications before using them in production.

1. Introduction

Onebip is a system that allows mobile phone subscribers to transfer pre-paid or post-paid credit from their SIM card to another user.

Through Onebip, mobile phone credit can be used to conduct financial transactions on a person-to-person (P2P) or person-to-merchant (P2M) basis for products/services other than voice and data services offered by mobile network operators.

In other words, Onebip turns mobile phone credit normally used for telecommunication purposes into electronic money that can be transferred among individuals or merchants.

Onebip payments can be received from both registered and non-registered users.

Registered users have access to all services offered by Onebip: store money into the account, receive payments from other users and withdraw funds in several ways.

Non-registered users can only send money to other users and have limited access to the functionalities of the system.

2. Integration Types

Onebip payments can be requested using 2 types of integrations: standard and express.

“Standard” integration is used when you want to collect money from a registered user.

With standard integrations you can determine the amount of money you want to receive using flexible price points.

“Express” integration is used when you want to collect money from a non-registered user directly from your website. Express payments can be used for a limited number of price points (see *Express Price Points*).

2.1 Standard

“Standard” integration is used when you want to collect money from a registered user.

This type of integration allows you to deploy the Onebip billing system on your website in a very short time without worrying about the complexity of managing a mobile phone-based payment on your servers.

Using this type of integration will allow you to request payments in a very flexible way.

In fact, thanks to the Onebip wallet, registered users can pay you any amount of money by topping up their onebip balance through their mobile phone credit.

Users can add funds to their Onebip wallet using a set of pre-determined price ranges that differ from country to country (see *Add Funds Options*)

As a result, you can request any amount (es. \$ 5.97, € 8.41, £ 15.75) just like any other online payment system, without the pricing limitations typically imposed by Mobile Network Operators.

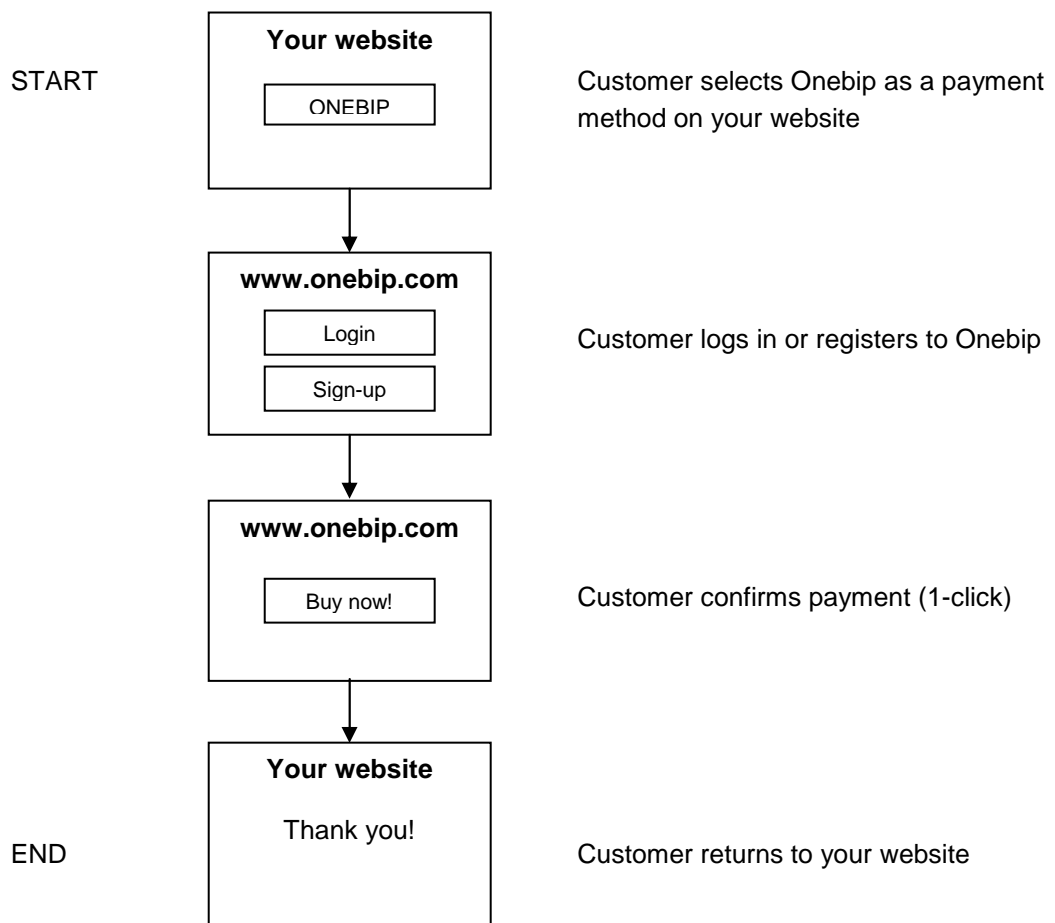
In standard integrations, also non-registered users can send you money by opening a onebip account. The onebip account creation requires a free of charge one-time registration process.

The registration is entirely managed by Onebip and does not require any action from your website/application.

Benefits of Standard payments:

- Flexible billing
- High price points available
- Low charge backs

Standard integration basic flow:



2.2 Express

Express integration is used when you want to collect money from a non-registered user.

This type of integration is ideal if you are looking for a full integration of onebip within your website. With Onebip Express you can request a payment directly from your pages without redirecting the user to an external page.

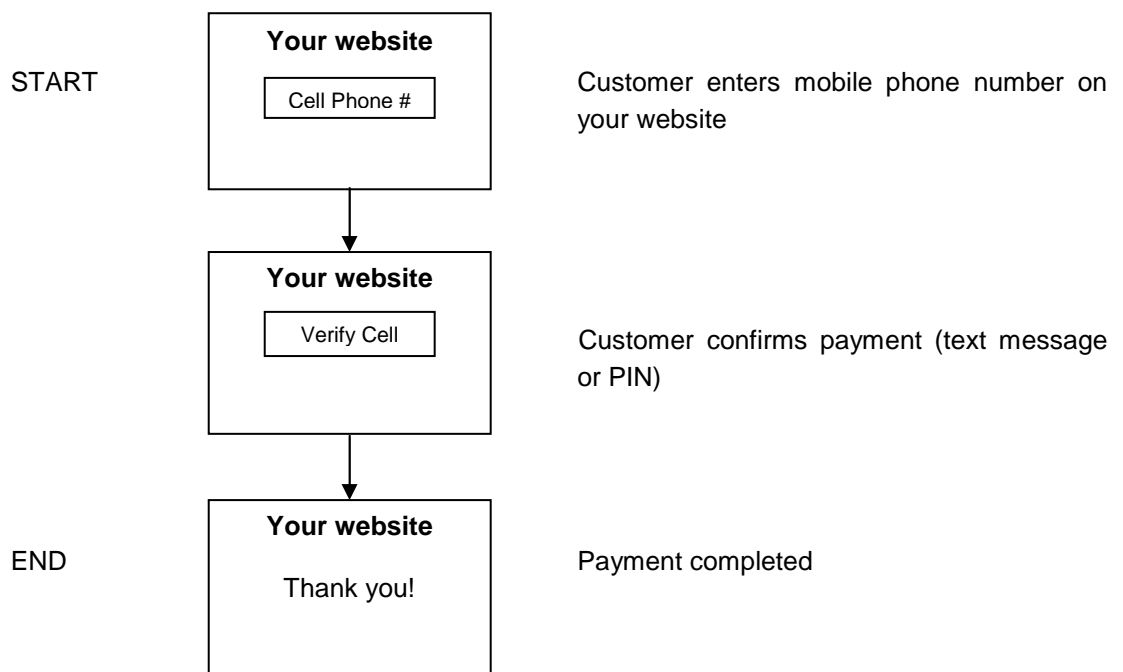
Each express payment has to be manually confirmed by the user by texting a message to a short code or by entering a PIN code into a web form. In both cases the flow is entirely managed by onebip and does not require any action from your website/application.

Express payments can be requested for a limited number of price points based on the country origin of the user (see *Express Price Points*)

Benefits of Onebip express:

- Full integration within your website
- Fast check-out process for non-registered users

Express integration basic flow:



3 One-time payments

With one-time payments your customers are billed only once for the amount of money you are requesting. Typically used for single items, one-time payments can be deducted from the payer's account, in case of a "standard" integration, or deducted directly from the mobile phone in case of "express" integration.

3.1 Constructing one-time payment requests

Linking to the Onebip payment page is achieved by using HTTP GET or POST requests to the following URL:

<https://www.onebip.com/otms/>

The following are case sensitive dynamic parameters you have to use in your payment query string to Onebip:

Name	Required?	Description	Length
username	Yes	The email address associated to your Onebip account.	255
description	Yes	Description of the item being sold. This will be displayed on the payment page and on the payment receipt.	255
price	Yes	End user price in cents/pence (local value added taxes included) as integer cents (actual amount * 100).	-
currency	Yes	Local currency code in ISO 4217 standard http://en.wikipedia.org/wiki/Currency_codes e.g. USD For allowable values, see Appendix "Currency Codes"	3
command	Yes	Determines the onebip integration type. Allowable values: standard_pay express_pay	50
country	No	Used only to restrict payments to users from a specific country. If no value is inserted the price will be automatically converted in the local currency of your customer Country code in ISO 3166 standard http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2 e.g. US For allowable values, see Appendix "Country Codes"	

country_disabled	No	Used to prevent users from specific countries to send you a payment. Typically used if you don't want to collect money from weak-currency countries Country code in ISO 3166 standard http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2 e.g. EG, CO, CL Include one or more country codes comma separated.	
item_code	No	Pass-through variable you can use to identify your internal item code for this payment. Note: This is not the Onebip item ID	64
lang	No	The language used on the payment page. Language code in ISO 639-1 http://en.wikipedia.org/wiki/List_of_ISO_639-1_codes e.g. en Default – English For allowable values, see Appendix “Language Codes”	2
return_url	No	The URL to which the payer's browser is redirected after completing the payment; for example, a URL on your site that displays a “Thank you for your payment” page. Default – The browser is redirected to a Onebip web page.	255
notify_url	No	The URL to which Onebip sends information about the transaction. For more information about notifications (see <i>Notification Parameters</i>). The parameters are passed using HTTP GET method.	255
cancel_url	No	A URL to which the payer's browser is redirected if payment is cancelled or failed by the merchant; for example, a URL on your website that displays a “Payment Cancelled” page.	255
remote_txid	No	Pass-through variable, a unique transaction reference number for the merchant system (your internal transaction ID).	64

custom	No	User-defined array of key-value pairs which will be passed through the system and returned in your <code>notify_url</code> and will be appended to your <code>return_url</code> Note: Your variable names cannot be the same as the one used by Onebip (see <i>Notification Parameters</i>)	max 10 variables (64 char)
customer_email	No	Payer's email address	255
customer_email_lock	No	Boolean value (true or false, 1 or 0) If true, the email address field on the payment page can't be edited manually from the user	5
customer_firstname	No	Payer's first name	50
customer_firstname_lock	No	Boolean value (true or false, 1 or 0) If true, the first name field on the payment page can't be edited manually from the user	5
customer_lastname	No	Payer's last name	50
customer_lastname_lock	No	Boolean value (true or false, 1 or 0) If true, the last name field on the payment page can't be edited manually from the user	5
customer_cell	No	Payer's cell phone number in international format with no leading e.g. 447700900999 Note: <code>customer_cell</code> must match <code>customer_country</code> or <code>country</code>	15
customer_cell_lock	No	Boolean value (true or false, 1 or 0) If true, the mobile phone number field on the payment page can't be edited manually from the user	5
customer_country	No	Payer's country code in ISO 3166 standard http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2 e.g. US For allowable values, see Appendix "Country Codes" Note: <code>customer_country</code> must match <code>country</code> , if used.	2

logo_url	No	The URL of the 360x45-pixel image displayed as your logo in the upper left corner of the Onebip payment pages. Supported formats: jpeg, gif, png	255
terms_url	No	The URL in which you have published special terms and conditions related to the item you are selling	255
debug	No	Boolean value (true or false, 1 or 0) If true, displays a verbose error message in case of integration failure. Note: Recommended for development purposes only.	5
debug_url	No	The URL in which you will be notified in case of integration failure. The call will include the error and the list of all parameters passed to the payment page	255

Note: When linking to the Onebip payment page, you can use a normal HTTP request or to use 128bit SSL encryption on our secure server. In both cases, the way to call the payment page is the same: if you want to use the secure server, just use HTTPS instead of HTTP in the payment page URL.

HTML Examples

The sample HTML code below illustrates a HTTP GET request with these features:

- Username: business@email.com
- Item name: 1000 game credits
- Item price: \$ 0.99 USD
- Return URL: <http://www.yoursite.com/thankyou.htm>
- Notify URL: <http://www.yoursite.com/notify.php>

https://www.onebip.com/otms/?command=standard_pay&username=business%40email.com&description=1000+game+credits&price=99¤cy=USD&return_url=http%3A%2F%2Fwww.yoursite.com%2Fthankyou.htm¬ify_url=http%3A%2F%2Fwww.yoursite.com%2Fnotify.php

An example of this communication using an HTTP POST request with optional parameters would be:

```
<form action="https://www.onebip.com/otms/" method="post" target="onebip">
<input type="hidden" name="command" value="standard_pay" />
<input type="hidden" name="username" value="business@email.com" />
<input type="hidden" name="description" value="1000 game credits" />
<input type="hidden" name="item_code" value="abc123" />
<input type="hidden" name="price" value="99" />
<input type="hidden" name="currency" value="USD" />
```

```

<input type="hidden" name="country" value="US" />
<input type="hidden" name="lang" value="en" />
<input type="hidden" name="return_url"
value="http://www.yoursite.com/thankyou.htm" />
<input type="hidden" name="cancel_url"
value="http://www.yoursite.com/cancel.htm" />
<input type="hidden" name="notify_url"
value="http://www.yoursite.com/notify.php" />
<input type="hidden" name="remote_txid" value="12QW34ER56TY" />
<input type="hidden" name="custom[your_variable1]" value="yourvalue1" />
<input type="hidden" name="custom[your_variable2]" value="yourvalue2" />
<input type="hidden" name="custom[foo]" value="bar" />
<input type="hidden" name="customer_email" value="user@email.com" />
<input type="hidden" name="customer_firstname" value="John" />
<input type="hidden" name="customer_lastname" value="Smith" />
<input type="hidden" name="customer_cell" value="17586584556" />
<input type="hidden" name="customer_country" value="US" />
<input type="hidden" name="logo_url" value="http://www.yoursite.com/logo.jpg" />
<input type="hidden" name="debug" value="true" />
<input type="image" name="submit"
src="http://www.onebip.com/tools/bts/btn04.gif" alt="Pay with Onebip" border="0"
/>
</form>

```

Below the URLs of the onebip buttons currently available:

<http://www.onebip.com/tools/bts/btn01.gif>
<http://www.onebip.com/tools/bts/btn02.gif>
<http://www.onebip.com/tools/bts/btn03.gif>
<http://www.onebip.com/tools/bts/btn04.gif>
<http://www.onebip.com/tools/bts/btn05.gif>
<http://www.onebip.com/tools/bts/btn06.gif>
<http://www.onebip.com/tools/bts/btn07.gif>
<http://www.onebip.com/tools/bts/btn08.gif>
<http://www.onebip.com/tools/bts/btn09.gif>
<http://www.onebip.com/tools/bts/btn10.gif>
<http://www.onebip.com/tools/bts/btn20.gif>

3.2 Processing the Onebip transaction response

The transaction response is to be executed by the merchant landing script to ensure the transaction is committed on the Onebip system before recording the customer payment as successful in the merchant database.

This guarantees no discrepancies between the merchant's transaction records and Onebip.

The following Onebip values must be confirmed as returned to the merchant's landing script:

Outcome	Description
OK	Payment accepted
ERROR: <i>Your error message</i>	Payment rejected

If the request has been handled correctly, your `notify_url` will have to output the word OK, upper case, in plain text format without any HTML code.

If your web page fails to respond OK, for instance if there is a system error, Onebip will place all notifications in a queue and retry them periodically until they are successfully delivered.

The retries will be performed every 5 minutes for the first hour, and hourly after that.

The total retry period for a payment notification is 24 hours.

During the retry period the payment will be stored in your onebip account as “Pending”.

If Onebip receives no outcome after the retry period the payment will be automatically cancelled and will be stored in your onebip account as “Failed”.

ERROR message should be returned in case of parameters mismatch in the `notify_url`, payment duplication or any other appropriate reason identified by your system.

All payment details including status (pending, failed, completed) and ERROR reasons are available in your onebip business account in the “Transactions History” section.

Note: In case of ERROR outcome or no outcome after the retry period, funds will not be deducted from the customer’s account and you will not be entitled to collect the payment

3.2.1 Notification Parameters

Onebip will notify you at the end of each successful payment. This is done with an HTTP GET request by our server to the `notify_url`, hosted on your servers.

To configure or change your site’s `notify_url`, you have to set the URL of a web page that receives the notification and takes appropriate action on your side.

The following are case sensitive dynamic parameters Onebip will notify you for each payment:

Parameter name	Always present?	Description
<code>notify</code>	Yes	Type of notification. For one-time payments is: <code>payment_single</code>
<code>payment_id</code>	Yes	This is a unique ID identifying the payment at Onebip. This ID will be shown in your panel and in your customers’ panel, and will be used to identify the payment. You may use this ID to prevent that the same payment is registered more than once on your side.
<code>country</code>	Yes	Country code in ISO 3166 standard http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2
<code>currency</code>	Yes	Local currency code in ISO 4217 standard http://en.wikipedia.org/wiki/Currency_codes
<code>price</code>	Yes	End user price in cents/pence (local value added taxes included). The billed amount is passed as integer cents (actual value * 100).

tax	Yes	Local value added taxes paid by the end user, if any (actual value * 100).
commission	Yes	Onebip commission fees for handling the transaction (actual value * 100). Note: Mobile Network Operator fees included
amount	Yes	Merchant net revenue for the transaction after local taxes and onebip fees (actual value * 100).
original_price	Yes	The original price set by the merchant before the onebip conversion in the local price of the buyer Note: Make sure that the amount notified is the same as the price amount passed in the payment request
original_currency	Yes	The original currency set by the merchant before the onebip conversion in the local currency of the buyer
remote_txid	No	Pass-through variable, a unique transaction reference number for the merchant system (your internal transaction ID).
item_code	No	Pass-through variable you can use to identify your internal item code for this payment.
custom	No	Any custom merchant parameters specified by the merchant in the transaction URL are returned here.
hash	No	Hash checksum of the merchant API Key and the Notification URL parameters.

PHP Example

```
<?php
// Report parameters

// Onebip parameters:
$payment_id      = $_REQUEST['payment_id'];
$country         = $_REQUEST['country'];
$currency        = $_REQUEST['currency'];
$price          = $_REQUEST['price'];
$tax             = $_REQUEST['tax'];
$commission      = $_REQUEST['commission'];
$amount         = $_REQUEST['amount'];
$original_price  = $_REQUEST['original_price'];
$original_currency = $_REQUEST['original_currency'];

// Your internal transaction ID and item code, if you use them:
$remote_txid    = $_REQUEST['remote_txid'];
$item_code      = $_REQUEST['item_code'];

// Your custom parameters, if you use them:
$your_variable1 = $_REQUEST['your_variable1'];
$your_variable2 = $_REQUEST['your_variable2'];
$foo            = $_REQUEST['foo'];
```

```

/*
...
you can use these parameters to make functions like
payment check, payment log, account enable ....
...
*/

if ($error) {
    echo 'ERROR: Your error message';
} else {
    echo 'OK'; // it is important you print "OK" in uppercase
}

?>

```

ASP Example

```

<%

' Report parameters

' Onebip parameters:
payment_id      = Request("payment_id")
country        = Request("country")
currency       = Request("currency")
price          = Request("price")
tax            = Request("tax")
commission     = Request("commission")
amount         = Request("amount")
original_price  = Request("original_price")
original_currency = Request("original_currency")

' Your internal transaction ID and item code, if you use them:
remote_txid    = Request("remote_txid")
item_code      = Request("item_code")

' Your custom parameters, if you use them:
your_variable1 = Request("your_variable1")
your_variable2 = Request("your_variable2")
foo            = Request("foo")

' You can use these parameters to make functions like
' payment check, payment log, account enable...

If error Then
    Response.Write("ERROR: Your error message")
Else
    Response.Write("OK")
    ' It is important that you print "OK" in upper case
End If

%>

```

3.2.2 MD5 Encryption of the Notification URL

MD5 hashing is an encryption method that ensures that URLs have not been manipulated or changed. This allows the integrity of the link in a data exchange between two parties to be checked.

MD5 is an abbreviation for “Message Digest Algorithm 5” and is a widely used cryptographic Hash function.

The length of the hash value produced by this hash function is 128 bits. The hash value is returned as a 32-digit hexadecimal number, which looks like this:

```
34048ce4cd069b624f6e021ba63ecde5
```

The MD5 hash verifies the integrity of the URL by comparing the MD5 hash created by the merchant system with a previously established hash made by the Onebip system. In this way, it can be determined whether the URL has been changed.

The primary purpose of this is to prevent the manipulation of price and currency.

The MD5 feature requires a secret key value (API Key) of the merchant’s choice that can be set under the “My Account” section.

A PHP example of how Onebip create the MD5 hash and append it as parameter `hash` to the Notification URL would be:

```
$basename = basename($notificationURL);
$hash = md5($ApiKey . $basename);
$notificationURL .= "&hash=" . $hash;
```

The PHP and ASP code below shows how standard and merchant parameters can be retrieved by the notification URL; the script can then use these settings to grant access, add credits, or perform any other action your integration requires.

As an added security measure, the notification URL should not be too obvious (e.g. <http://www.yoursite.com/paymentnotification.php> might be an easy guess)

PHP Example

```
<?php
// Check MD5 hash - anti-fraud measure

if (isset($_REQUEST['hash'])) {
    $my_api_key = 'My Secret Key'; // stored in your account settings
    $basename = basename($_SERVER['REQUEST_URI']);
    $pos = strpos($basename, "&hash");
    $basename_without_hash = substr($basename, 0, $pos);
    $my_hash = md5($my_api_key . $basename_without_hash);

    if ($my_hash != $_REQUEST['hash']) {
        log("ERROR: this isn't a valid Onebip notification!");
        // Note: "log()" sample function
        echo "ERROR: Invalid hash code";
        exit();
    }
}

echo 'OK'; // it is important you print "OK" in uppercase

?>
```

ASP Example

```

<%
' Check MD5 hash - anti-fraud measure

dim hash, my_api_key, basename, basenameexpl, pos, basename_without_hash,
my_hash

hash = Request("hash")
If hash <> "" Then
    my_api_key = "My Secret Key"
    ' stored in your account settings
    basename = Request.ServerVariables("SCRIPT_NAME")
    basenameexpl = split(basename, "/")
    basename = basenameexpl(ubound(basenameexpl))
    pos = InStr(request.querystring, "&hash")
    basename_without_hash = basename&left(Request.querystring, pos-1)
    my_hash = Md5(my_api_key&basename_without_hash)
    ' Note: "Md5()" sample function
    If my_hash <> hash Then
        log("ERROR: this isn't a valid Onebip notification!" )
        ' Note: "Log()" sample function
        Response.Write("ERROR: Invalid hash code")
        Response.End()
    End If
End If

Response.Write("OK")
' It is important that you print "OK" in upper case

%>

```

3.3 Return URL

Onebip will append all the pass-through parameters you have specified in the transaction URL to your `return_url`.

Passthrough parameters in the `return_url` page can be used in many ways, for example to pass an internal userID through the payment process and redirect the customer to its account on your site, or to build customised thank you pages. The `return_url` page can also route your user to a specific download, using the `notify_url` to authorize it.

The `return_url` page could be used to host the scripts needed to grant the permissions for the user to access a specific product/service.

In case of no `return_url` parameter, the user will be redirected to a generic Onebip thank you page with the payment summary.

Note: We strongly suggest any permission or other action needed to deliver the service or product to the final user is *not* triggered by the `return_url` page but by the `notify_url`

4 Subscriptions

The subscription module is used when Onebip is required to make regular automatic charges to a customer (e.g. daily, weekly, monthly or annual charges).

Due to some technical limitations imposed by Mobile Network Operators, subscription-based payments cannot benefit from the pricing flexibility offered by the onebip standard integration.

As a result the `price` parameter for subscriptions cannot be a flexible value but has to respect a set of pre-determined price points which differ from country to country.

If you use subscription-based billing services, a new notification will be sent for each successful billing (see *Notification Parameters*).

For a list of all price points currently available and the list of subscription enabled countries, please refer to *Subscription Price Points (Annex)*

4.1 Constructing subscription requests

In addition to the parameters requested for one-time payments, in order to request a subscription the following mandatory parameters are required:

Name	Required?	Description	Length
<code>country</code>	Yes	Used only to restrict payments to users from a specific country. If no value is inserted the <code>price</code> will be automatically converted in the local currency of your customer Country code in ISO 3166 standard http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2 e.g. US For allowable values, see Appendix "Country Codes"	2
<code>frequency</code>	Yes	Used for recurring payments only. Defines the interval in between each payment in days e.g daily=1 weekly=7 monthly=30	3

Note: All other mandatory parameters requested for one-time payments have to be included also when constructing a subscription request (see *Constructing one-time payment requests*)

4.2 Processing the Onebip transaction response

The transaction response is to be executed by the merchant landing script to ensure the transaction is committed on the Onebip system before recording the customer payment as successful in the merchant database.

This guarantees no discrepancies between the merchant's transaction records and Onebip.

The following Onebip values must be confirmed as returned to the merchant's landing script:

Outcome	Description
OK	Payment accepted
ERROR: <i>Your error message</i>	Payment rejected

If the request has been handled correctly, your `notify_url` will have to output the word OK, upper case, in plain text format without any HTML code.

If your web page fails to respond OK, for instance if there is a system error, Onebip will place all notifications in a queue and retry them periodically until they are successfully delivered.

The retries will be performed every 5 minutes for the first hour, and hourly after that.

The total retry period for a payment notification is 24 hours.

During the retry period the payment will be stored in your onebip account as "Pending".

If Onebip receives no outcome after the retry period the payment will be automatically cancelled and will be stored in your onebip account as "Failed".

ERROR message should be returned in case of parameters mismatch in the `notify_url`, payment duplication or any other appropriate reason identified by your system.

All payment details including status (pending, failed, completed) and ERROR reasons are available in your onebip business account in the "Transactions History" section.

Note: Onebip will automatically attempt to bill the user based on the frequency of the subscription. All successful attempts will be notified to your server like any other onebip payment.

4.2.1 Notification Parameters

Onebip will notify you at the end of each successful payment. This is done with an HTTP GET request by our server to the `notify_url`, hosted on your servers.

To configure or change your site's `notify_url`, you have to set the URL of a web page that receives the notification and takes appropriate action on your side.

In addition to the case sensitive dynamic parameters notified for one-time payments, for subscriptions the following parameters will be included:

Parameter name	Always present?	Description
<code>notify</code>	Yes	Type of notification. For subscriptions allowable values are: <code>payment_subscription</code> <code>unsubscription</code> Note: <code>payment_subscription</code> notifies the activation of a subscription and <code>unsubscription</code> the cancellation of a subscription
<code>subscription_id</code>	Yes	This is a unique ID identifying a subscription. This ID will be passed to all recurring payments associated to that particular subscription or when notifying an unsubscription.

Subscription activation example:

http://www.yoursite.com/onebip_notify.php?notify=payment_subscription&subscription_id=1234567

Subscription cancellation example:

http://www.yoursite.com/onebip_notify.php?notify=unsubscription&subscription_id=1234567

4.2.2 MD5 Encryption of the Notification URL

MD5 hashing is an encryption method that ensures that URLs have not been manipulated or changed. This allows the integrity of the link in a data exchange between two parties to be checked.

MD5 is an abbreviation for "Message Digest Algorithm 5" and is a widely used cryptographic Hash function.

The length of the hash value produced by this hash function is 128 bits. The hash value is returned as a 32-digit hexadecimal number, which looks like this:

```
34048ce4cd069b624f6e021ba63ecde5
```

The MD5 hash verifies the integrity of the URL by comparing the MD5 hash created by the merchant system with a previously established hash made by the Onebip system. In this way, it can be determined whether the URL has been changed.

The primary purpose of this is to prevent the manipulation of price and currency.

The MD5 feature requires a secret key value (API Key) of the merchant's choice that can be set under the "My Account" section.

A PHP example of how Onebip create the MD5 hash and append it as parameter `hash` to the Notification URL would be:

```
$basename = basename($notificationURL);
$hash = md5($ApiKey . $basename);
$notificationURL .= "&hash=" . $hash;
```

The PHP and ASP code below shows how standard and merchant parameters can be retrieved by the notification URL; the script can then use these settings to grant access, add credits, or perform any other action your integration requires.

As an added security measure, the notification URL should not be too obvious (e.g. <http://www.yoursite.com/paymentnotification.php> might be an easy guess)

PHP Example

```
<?php
// Check MD5 hash - anti-fraud measure
if (isset($_REQUEST['hash'])) {
    $my_api_key = 'My Secret Key'; // stored in your account settings
    $basename = basename($_SERVER['REQUEST_URI']);
    $pos = strrpos($basename, "&hash");
    $basename_without_hash = substr($basename, 0, $pos);
    $my_hash = md5($my_api_key . $basename_without_hash);

    if ($my_hash != $_REQUEST['hash']) {
        log("ERROR: this isn't a valid Onebip notification!");
        // Note: "log()" sample function
        echo "ERROR: Invalid hash code";
        exit();
    }
}

echo 'OK'; // it is important you print "OK" in uppercase
?>
```

ASP Example

```

<%
' Check MD5 hash - anti-fraud measure

dim hash, my_api_key, basename, basenameexpl, pos, basename_without_hash,
my_hash

hash = Request("hash")
If hash <> "" Then
    my_api_key = "My Secret Key"
    ' stored in your account settings
    basename = Request.ServerVariables("SCRIPT_NAME")
    basenameexpl = split(basename, "/")
    basename = basenameexpl(ubound(basenameexpl))
    pos = InStr(request.querystring, "&hash")
    basename_without_hash = basename&left(Request.querystring, pos-1)
    my_hash = Md5(my_api_key&basename_without_hash)
    ' Note: "Md5()" sample function
    If my_hash <> hash Then
        log("ERROR: this isn't a valid Onebip notification!" )
        ' Note: "Log()" sample function
        Response.Write("ERROR: Invalid hash code")
        Response.End()
    End If
End If

Response.Write("OK")
' It is important that you print "OK" in upper case

%>

```

4.3 Return URL

Onebip will append all the pass-through parameters you have specified in the transaction URL to your `return_url`.

Pass-through parameters in the `return_url` page can be used in many ways, for example to pass an internal userID through the payment process and redirect the customer to its account on your site, or to build customised thank you pages. The `return_url` page can also route your user to a specific download, using the `notify_url` to authorize it.

The `return_url` page could be used to host the scripts needed to grant the permissions for the user to access a specific product/service.

In case of no `return_url` parameter, the user will be redirected to a generic Onebip thank you page with the payment summary.

Note: We strongly suggest any permission or other action needed to deliver the service or product to the final user is *not* triggered by the `return_url` page but by the `notify_url`

For subscriptions the user will be redirected to the `return_url` only for the first payment of the subscription. The following payments will be only notified to your `notify_url`

5 API Operations

1.1 Download Reports

Merchants can download profit reports from Onebip automatically using the Onebip API. This API method is currently in beta state and may be altered in future versions to meet the requirements of our partners.

Merchants can obtain information about Onebip transactions simply using the Onebip Web Service.

In order to retrieve information through our web service you will need to configure your client as follows:

SOAP SERVER URL: `https://www.onebip.com/api/soap/`

WSDL URL: `https://www.onebip.com/api/soap/?wsdl`

Currently the web service supports only the following method:

Name: *getPayments*

Input: An array of parameters as described below:

Parameter name	Type	Description
username	String	The email address associated to your Onebip account.
password	String	Your Onebip password.
from_date_time	String	Start Date and Time of the report in YYYY-MM-DD HH:MM:SS format (UTC)
to_date_time	String	End Date and Time of the report in YYYY-MM-DD HH:MM:SS format (UTC)

Output: An array of objects with the following data members:

Data members	Type	Description
payment_id	Int	This is a unique ID identifying the payment at Onebip. This ID will be shown in your panel and in your customers' panel, and will be used to identify the payment.
date_time	String	Payment Date and Time (UTC)
country	String	Country code in ISO 3166 standard http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2
currency	String	Local currency code in ISO 4217 standard http://en.wikipedia.org/wiki/Currency_codes
price	Float	End user price in local currency (local value added taxes included).
tax	Float	Local value added taxes paid by the end user, if any.
commission	Float	Onebip commission fees for handling the transaction

Note: Mobile Network Operator fees included

amount	Float	Merchant net revenue for the transaction after local taxes and onebip fees.
remote_txid	String	Pass-through variable, a unique transaction reference number for the merchant system (your internal transaction ID)
item_code	String	Pass-through variable you can use to identify your internal item code for this payment.
custom	Struct	Any custom merchant parameters specified by the merchant in the transaction URL are returned here.

In case of error a *SOAP_Fault* object will be returned containing the error description. **Currently, the maximum limit of records retrievable with a single call is 200.**

1.2 Cancel subscription

Merchants can stop a subscription automatically using the Onebip API

Web service URL: `http(s)://www.onebip.com/api/subscription/`

Input: GET parameters as described below:

Parameter name	Description
username	The Onebip ID or the email address associated to your Onebip account.
password	Your Onebip password.
action	The type of request you want to perform. To cancel a subscription use: unsubscribe
subscription_id	The ID of the subscription that you want to cancel.

Output: A plain/text message (no HTML)

Outcome	Description
OK	Subscription cancelled successfully
ERROR: error description	Cancellation request failed

Example:

http://www.onebip.com/api/subscription/?username=name@mail.com&password=abcd&action=unsubscribe&subscription_id=1234567

6 Appendix

6.1 Nation Codes

The table below shows the list of countries currently supported by Onebip in ISO 4217 standard:

Nation	Code	Nation	Code
Argentina	AR	Lithuania	LT
Australia	AU	Luxemburg	LU
Austria	AT	Malaysia	MY
Belgium	BE	Mexico	MX
Bosnia and Herzegovina	BA	Morocco	MA
Bulgaria	BG	Netherlands	NL
Canada	CA	Norway	NO
Chile	CL	Peru	PE
Colombia	CO	Poland	PL
Croatia	HR	Portugal	PT
Czech Republic	CZ	Romania	RO
Denmark	DK	Russia	RU
Ecuador	EC	Serbia	RS
Egypt	EG	Singapore	SG
Estonia	EE	Slovenia	SI
Finland	FI	South Africa	ZA
France	FR	Spain	ES
Germany	DE	Sweden	SE
Greece	GR	Switzerland	CH
Hungary	HU	Taiwan, Province of China	TW
Indonesia	ID	Thailand	TH
Ireland	IE	Turkey	TR
Italy	IT	Ukraine	UA
Kazakhstan	KZ	United Kingdom	UK
Latvia	LV	United States	US

For updates please refer to http://en.wikipedia.org/wiki/ISO_3166-1

6.2 Currency codes

The table below shows the list of currencies currently supported by Onebip in ISO 4217 standard:

Currency	Code	Currency	Code
Australian Dollar	AUD	Lithuanian Lita	LTL
Bosnian Marka	BAM	Latvian Lat	LVL
Bulgarian Lev	BGN	Mexico Peso	MXN
Brazilian Real	BRL	Malaysian Ringgit	MYR
Brunei Dollar	BND	Norwegian Krone	NOK
Canadian Dollar	CAD	New Zealand Dollar	NZD
Swiss Franc	CHF	Philippine Peso	PHP
Chinese Yuan	CNY	Polish Zloty	PLN
Czech Koruna	CZK	Romanian New Leu	RON
Danish Krone	DKK	Serbian Dinar	RSD
Egyptian Pound	EGP	Russian Rouble	RUB
Estonian Kroon	EEK	Swedish Krona	SEK
Euro	EUR	Singapore Dollar	SGD
British Pound	GBP	Thai Baht	THB
Hong Kong Dollar	HKD	New Turkish Lira	TRY
Croatian Kuna	HRK	Taiwan Dollar	TWD
Hungarian Forint	HUF	UAE Dirham	AED
Indonesian Rupiah	IDR	Ukraine Hryvnia	UAH
Indian Rupee	INR	US Dollar	USD
Japanese Yen	JPY	Saudi Arabian Ryal	SAR
Korean Won	KRW	South African Rand	ZAR
Kazakhstan Tenge	KZT	Vietnam Dong	VND

For updates please refer to http://en.wikipedia.org/wiki/Currency_codes

6.3 Language Codes

The table below shows the list of languages currently supported by Onebip in ISO 639-1 standard:

Language Name	Native Name	Code
Bulgarian	Български	bg
Bosnian	Bosanski	bs
Croatian	Hrvatski	hr
Danish	Dansk	da
German	Deutsch	de
English	English	en
Spanish	Español	es
Estonian	Eesti keel	et
Filipino	Filipino	fil
French	Français	fr
Indonesian	Bahasa Indonesia	id
Italian	Italiano	it
Latvian	Latviešu valoda	lv
Lithuanian	Lietuvių kalba	lt
Macedonian	Makedonski	mk
Dutch	Nederlands	nl
Norwegian	Norsk	no
Polish	Polski	pl
Portuguese (Brasil)	Português (Brasil)	pt_BR
Portuguese (Portugal)	Português (Portugal)	pt_PT
Romanian	Română	ro
Finnish	Suomi	fi
Slovak	Slovenčina	sk
Slovene	Slovenščina	sl
Swedish	Svenska	sv
Turkish	Türkçe	tr
Czech	Čeština	cs
Greek	Ελληνικά	el
Serbian	Srpski	Sr
Russian	Русский	Ru
Ukrainian	Українська	Uk
Hungarian	Magyar	Hu
Hebrew	עברית	He
Arabic	العربية	Ar
Thai	ภาษาไทย	Th

Chinese	中文 (简体)	zh_CN
Japanese	日本語	Ja
Korean	한국어	Ko

For updates please refer to http://en.wikipedia.org/wiki/List_of_ISO_639-1_codes

6.4 Subscription-based price points (VAT incl.)

The table below shows the list of subscription-enabled countries and price points currently supported by Onebip:

Nation	Currency	Price 1	Price 2	Price 3	Price 4
Australia	AUD	6,60	-	-	-
Belgium	EUR	4,00	-	-	-
Bosnia Herzegovina	BAM	2,34	-	-	-
Canada	CAD	5,00	-	-	-
Croatia	HRK	6,10	-	-	-
Czech Republic	CZK	79,00	-	-	-
Denmark	DKK	30,00	60,00	-	-
Finland	EUR	5,00	-	-	-
France	EUR	3,00	-	-	-
Germany	EUR	2,99	4,99	6,99	9,99
Hungary	HUF	1.000,00	1.250,00	1.875,00	-
Indonesia	IDR	8.000,00	10.000,00	15.000,00	-
Ireland	EUR	2,00	-	-	-
Italia	EUR	2,00	4,00	5,00	10,00
Latvia	LVL	0,95	3,00	-	-
Lithuania	LTL	5,00	10,00	-	-
Malaysia	MYR	5,00	7,00	10,00	-
Netherlands	EUR	1,50	-	-	-
Norway	NOK	20,00	30,00	50,00	60,00
Serbia	RSD	236,00	472,00	-	-
Singapore	SGD	3,00	5,00	8,00	10,00
South Africa	ZAR	30,00	50,00	-	-
Sweden	SEK	10,00	20,00	30,00	50,00
Taiwan	TWD	30,00	50,00	100,00	-
Thailand	THB	25,00	50,00	100,00	150,00
United Kingdom	GBP	1,50	3,00	5,00	10,00
United States	USD	2,99	3,99	5,99	9,99

6.5 Onebip “add funds” price points (VAT incl.)

Nation	Currency	Price 1	Price 2	Price 3	Price 4
Argentina	ARS	4,84	9,68	19,36	29,04
Australia	AUD	6,60	13,20	19,80	26,40
Austria	EUR	1,99	4,99	9,99	19,99
Belgium	EUR	4,00	8,00	12,00	16,00
Bosnia Herzegovina	BAM	2,34	4,68	7,02	9,36
Bulgaria	BGN	2,40	4,80	7,20	9,60
Canada	CAD	5,00	10,00	15,00	20,00
Chile	CLP	700,00	1.400,00	2.800,00	4.200,00
Colombia	COP	4.522,00	9.044,00	13.566,00	18.088,00
Croatia	HRK	6,10	12,20	18,30	24,40
Czech Republic	CZK	79,00	158,00	237,00	316,00
Denmark	DKK	30,00	60,00	120,00	150,00
Ecuador	USD	1,45	2,90	5,80	8,70
Estonia	EEK	35,00	50,00	100,00	150,00
Finland	EUR	5,00	10,00	15,00	20,00
France	EUR	3,00	6,00	9,00	12,00
Germany	EUR	2,99	4,99	6,99	9,99
Greece	EUR	1,23	3,62	7,24	10,86
Hungary	HUF	1.000,00	1.250,00	1.875,00	3.750,00
Hong Kong	HKD	30,00	60,00	90,00	120,00
Indonesia	IDR	15.000,00	30.000,00	45.000,00	60.000,00
Ireland	EUR	2,00	4,00	6,00	8,00
Italia ¹	EUR	1,00	3,00	5,00	10,00
Kazakhstan	KZT	75,00	375,00	535,00	1.070,00
Latvia	LVL	3,00	6,00	9,00	12,00
Lithuania	LTL	10,00	20,00	30,00	40,00
Luxemburg	EUR	3,00	6,00	9,00	12,00
Malaysia	MYR	10,00	20,00	30,00	40,00
Mexico	MXN	14,95	29,90	59,80	89,70
Morocco	MAD	24,00	48,00	72,00	96,00
Netherlands	EUR	1,50	3,00	4,50	6,00
Norway	NOK	30,00	60,00	120,00	180,00
Peru	PEN	7,00	14,00	17,50	21,00
Poland	PLN	10,98	23,18	46,36	69,54

¹ Flexible billing does not apply to Italy. Onebip will automatically select the closest price point available.

Portugal	EUR	2,00	4,00	6,00	8,00
Romania	EUR	2,60	5,20	7,80	10,40
Russia	RUB	200,00	400,00	600,00	800,00
Serbia	RSD	236,00	472,00	944,00	1.416,00
Singapore	SGD	5,00	10,00	20,00	30,00
Slovenia	EUR	0,99	1,99	3,98	5,97
South Africa	ZAR	30,00	50,00	100,00	150,00
Spain	EUR	1,42	4,06	7,08	14,16
Sweden	SEK	25,00	50,00	100,00	150,00
Switzerland	CHF	5,00	10,00	15,00	20,00
Taiwan	TWD	50,00	100,00	200,00	300,00
Thailand	THB	100,00	150,00	300,00	450,00
Turkey	TRY	5,00	10,00	20,00	40,00
Ukraine	UAH	30,00	60,00	90,00	120,00
United Arab Emirates	AED	10,00	20,00	30,00	40,00
United Kingdom	GBP	3,00	5,00	10,00	15,00
United States	USD	2,99	3,99	5,99	9,99

6.6 Express price points (VAT incl.)

The table below shows the list of price points available for the express integration:

Nation	Currency	Price 1	Price 2	Price 3	Price 4
Argentina	ARS	4,84	-	-	-
Australia	AUD	6,60	-	-	-
Austria	EUR	1,99	4,99	9,99	19,99
Belgium	EUR	4,00	-	-	-
Bosnia Herzegovina	BAM	2,34	-	-	-
Bulgaria	BGN	2,40	-	-	-
Canada	CAD	5,00	-	-	-
Chile	CLP	700,00	-	-	-
Colombia	COP	4.522,00	-	-	-
Croatia	HRK	6,10	-	-	-
Czech Republic	CZK	79,00	-	-	-
Denmark	DKK	30,00	60,00	-	-
Ecuador	USD	1,45	-	-	-
Estonia	EEK	15,20	35,60	50,80	-
Finland	EUR	5,00	-	-	-
France	EUR	3,00	-	-	-
Germany	EUR	2,99	4,99	6,99	9,99

Greece	EUR	1,23	3,62	-	-
Hungary	HUF	1.000,00	1.250,00	1.875,00	-
Hong Kong	HKD	30,00	-	-	-
Indonesia	IDR	8.000,00	10.000,00	15.000,00	-
Ireland	EUR	2,00	-	-	-
Italia ²	EUR	3,00	5,00	7,00	10,00
Kazakhstan	KZT	75,00	375,00	535,00	-
Latvia	LVL	0,95	3,00	-	-
Lithuania	LTL	5,00	10,00	-	-
Luxemburg	EUR	3,00	-	-	-
Malaysia	MYR	5,00	7,00	10,00	-
Mexico	MXN	19,72	59,16	78,88	118,32
Morocco	MAD	24,00	-	-	-
Netherlands	EUR	1,50	-	-	-
Norway	NOK	20,00	30,00	50,00	60,00
Peru	PEN	3,50	-	-	-
Poland	PLN	10,98	23,18	-	-
Portugal	EUR	2,00	-	-	-
Romania	EUR	2,60	-	-	-
Russia	RUB	200,00	-	-	-
Serbia	RSD	236,00	472,00	-	-
Singapore	SGD	3,00	5,00	8,00	10,00
Slovenia	EUR	0,99	1,99	-	-
South Africa	ZAR	30,00	50,00	-	-
Spain	EUR	1,42	4,06	7,08	-
Sweden	SEK	10,00	20,00	30,00	50,00
Switzerland	CHF	3,00	5,00	-	-
Taiwan	TWD	30,00	50,00	100,00	-
Thailand	THB	25,00	50,00	100,00	150,00
Turkey	TRY	5,00	10,00	15,00	20,00
Ukraine	UAH	30,00	-	-	-
United Arab Emirates	AED	5,00	10,00	-	-
United Kingdom	GBP	1,50	3,00	5,00	10,00
United States	USD	2,99	3,99	5,99	9,99

² Other express price points available upon request