

Paynow – Link and Button Generation Guide

A guide outlining how to programmatically create clickable links and buttons to generate payment in Paynow, as well as receive updates via a notification URL post

Version 1.0.1
15 August 2014

Contents

Create Payment Request Buttons for E-mail or Web Page	2
How to Create a Simple Payment Request Button.....	2
How to Create an Advanced Payment Request Button.....	4
How the Notification, Success & Cancel URLs Work	7
POST Data.....	7
Verifying the Hash Value	8
URL Safe Base 64 Encoding	8

Create Payment Request Buttons for E-mail or Web Page

How to Create a Simple Payment Request Button

Using this functionality, you can send an e-mail (or put a button on a web page) requesting money from anybody with an email address and get notification as soon as they make payment.

The button or link you create should have a hyperlink to the following URL:

<https://www.paynow.co.zw/payment/link/<customer-email>?q=<arguments>>

The parameter denoted by **<customer-email>** is optional. If specified, Paynow will anonymously log the customer in using the specified address (or present them with a log in page if they already have a Paynow account)

The parameter denoted by **<arguments>** above should be replaced by a **URL encoded** and then **Base 64 encoded** (See [URL Safe Base 64 Encoding](#)) string of the following options, each option separated by an ampersand (&) character:

Argument	Name	Data Type	Description	Default	Example
Merchant E-mail	search	Text	The Paynow e-mail address of the merchant receiving payment	n/a	company@gmail.com
Payment Amount	amount	Numeric	The amount that is to be paid by the customer (see more below)	0.00	23.65
Payment Reference	reference	Text	The merchant's payment reference	(blank)	INV-12345
Locked	l	Numeric (1 or 0)	Whether or not the amount and reference fields will be editable by the customer	0	1

*N.B. If the "Locked" argument is omitted or specified as 0 (zero) then the customer will be able to amend the **amount** and **reference** values when they arrive at Paynow. This may be useful in cases where a donation amount (or any unspecified amount) is to be paid by the customer.*

Due to the simplicity of this integration it is possible, even if the payment is marked as locked, that the customer can change the amount and reference. It is important the merchant checks these values after receiving payment. If you wish to integrate in a way that it is impossible for the customer to edit the amount and reference please see the 3rd party shopping cart or link integration.

Example:

Step 1 – Gather the information you want user to see...

Merchant e-mail	company@gmail.com
Payment amount (optional)	12.50
Payment reference (optional)	ABC123
Locked (optional)	1

Step 2 – URL encode each argument...

Merchant e-mail	company%40gmail.com
Payment amount	12.50
Payment reference	ABC123
Locked	1

Step 3 – Construct the arguments into key/value pairs...

search=**company%40gmail.com**&amount=**12.50**&reference=**ABC123**&l=**1**

Step 4 – Base 64 encode the arguments (See [URL Safe Base 64 Encoding](#))

c2VhcmNoPWNvbXBhbmlAZ21haWwuY29tJmFtb3VudD0xMi41MCZyZWZlcmVuY2U9QUJDMTIzJmw9MQ==

Step 5 – URL encode the arguments (See [URL Safe Base 64 Encoding](#))

c2VhcmNoPWNvbXBhbmlAZ21haWwuY29tJmFtb3VudD0xMi41MCZyZWZlcmVuY2U9QUJDMTIzJmw9MQ%3D%3D

Step 6 – Create the URL of the link, optionally including the customer's email address before the arguments...

<https://www.paynow.co.zw/payment/link/customer@gmail.com?q=c2VhcmNoPWNvbXBhbmlAZ21haWwuY29tJmFtb3VudD0xMi41MCZyZWZlcmVuY2U9QUJDMTIzJmw9MQ%3D%3D>

How to Create an Advanced Payment Request Button

An advanced payment request button will allow the customer to enter additional details about the product or service they are purchasing. For the following tutorial, you will need to first create a "Custom Button Template" in the merchant's Paynow account. You can find this [here](#)

Once you've created a custom button template in Paynow, make note of its Integration ID as this will be required when creating the button or link URL.

The button or link you create should have a hyperlink to the following URL:

<https://www.paynow.co.zw/payment/billpaymentlink/<customer-email>?q=<arguments>>

The parameter denoted by **<customer-email>** is optional. If specified, Paynow will anonymously log the customer in using the specified address (or present them with a log in page if they already have a Paynow account)

The parameter denoted by **<arguments>** above should be replaced by a **URL encoded** and then **Base 64 encoded** (See [URL Safe Base 64 Encoding](#)) string of the following options, each option separated by an ampersand (&) character:

Argument	Name	Data Type	Description	Default	Example
ID	id	Numeric	The Integration ID number of the custom button template on Paynow (see above)	n/a	1046
Payment Amount	amount	Numeric	The amount to be paid by the customer (if "Use Quantity" is ticked, this will be the unit price)	0.00	25.00
Quantity	quantity	Numeric	The quantity being paid for by the customer (unless "Use Quantity" is ticked, this will be ignored)	1.00	2.36
Locked	l	Numeric (1 or 0)	Whether or not the input fields will be editable by the customer	0	1
Fields	f1, f2, f3 ... etc.	Text	Any extra fields you have added to the template, in the order they appear on the custom button template on Paynow	(blank)	(see below)

*N.B. If the "Locked" argument is omitted or specified as 0 (zero) then the customer will be able to amend the **amount, quantity** and **any other input field** values when they arrive at Paynow. This may be useful in cases where the customer's requirements are not pre-determined e.g. the quantity, item colour, caption and size of a t-shirt*

Due to the simplicity of this integration it is possible, even if the payment is marked as locked, that the customer can change the values. It is important the merchant checks these values after receiving payment. If you wish to integrate in a way that it is impossible for the customer to edit the amount and reference please see the 3rd party shopping cart or link integration.

Example:

This example uses a custom button template where three extra fields were added:

1. Item Colour (Select List)
2. Caption (Text)
3. Size (Text + Numeric)

Step 1 – Setup the Custom Button Template in Paynow and make note of its integration ID number.

1046

Step 2 – Make note of the order of any extra fields you’ve added to your Custom Button Template in Paynow, and the default values you’d like them to have (if any). They will be inserted into your arguments as f1, f2, f3 ... etc. in the order in which they have been setup in the template.

Payment amount (optional)	75.50
f1 (optional)	Red
f2 (optional)	Pay when? Paynow!
f3 (optional)	32
Locked (optional)	1

Step 3 – URL encode each argument...

Payment amount (optional)	75.50
f1 (optional)	Red
f2 (optional)	Pay+when%3F+Paynow%21
f3 (optional)	32
Locked (optional)	1

Step 4 – Construct the arguments into key/value pairs...

id=1046&amount=75.50&f1=Red&f2= Pay+when%3F+Paynow%21&f3=32&l=1

Step 4 – Base 64 encode the arguments (See [URL Safe Base 64 Encoding](#))

aWQ9MTA0NiZhbW91bnQ9NzUuNTAmZjE9UmVkJmYyPSBQYXkrd2hlbiUzRitQYXlub3cIMjEmZjM9MzI
mbD0x

Step 5 – URL encode the arguments (See [URL Safe Base 64 Encoding](#))

aWQ9MTA0NiZhbW91bnQ9NzUuNTAmZjE9UmVkJmYyPSBQYXkrd2hlbiUzRitQYXlub3cIMjEmZjM9MzI
mbD0x

Step 6 – Create the URL of the link, optionally including the customer’s email address before the arguments

<https://www.paynow.co.zw/payment/billpaymentlink/customer@gmail.com?q=aWQ9MTA0NiZhbW91bnQ9NzUuNTAmZjE9UmVkJmYyPSBQYXkrd2hlbiUzRitQYXlub3cIMjEmZjM9MzImbD0x>

How the Notification, Success & Cancel URLs Work

When using a **Custom Button Template** to create an advanced payment request button, you have the option of entering a URL of your own that Paynow can post data to (by means of a form POST) to let you know about status updates, payment success or payment cancellation respectively.

The post is sent to the Notification URL, the client's browser (with no extra information) will be redirected back to the Success or Cancel URL depending on transaction status.

POST Data

The data posted by Paynow to the Notification URL will include the following fields:

Field	Description	Data Type	Example
Paynow_Reference	The Paynow transaction reference	Text	40222
Customer_Name	The name of the customer	Text	John Smith
Customer_Email	The customer's email address	Text	customer@gmail.com
Customer_Phone	The customer's phone number	Text	0733123456
Transaction_Amount	The transaction amount	Numeric	25.00
Amount_Paid	The amount debited from the customer's account	Numeric	26.30
Hash	A hash of the data in the form post (see Verifying the Hash Value for more information)	Text	81D4957A0EB18F8079D33E5 C0AF0F0F604D6F127F3C9EC9 0D09FDB9001E1197CB0B276 38348BD9FDB1ED72DA60830 6F7E050C9445BFA08AC1A71 F83F500CA59C
(extra fields)	A list of any extra fields you may have set up in the custom button template	Text	(see table below)

Any extra fields that were setup in the Custom Button Template will be included in the form post with their key and value, If the name contains a space it will be replaced with an underscore _ for example:

Field	Value
Item_Colour	Red
Caption	Pay when? Pay now!
Size	32

Verifying the Hash Value

It is important for the developer to re-generate a hash of the posted data and compare it to the hash supplied in the form post to ensure that the data was in fact posted from Paynow (otherwise it is possible that the data may have been spoofed by a 3rd party)

You can find the Integration Key for the payment method in Paynow by clicking edit on the template. To generate a hash value for the message follow the steps:

1. Concatenate the key plus the value with no space or separator in the same order they appear in the post.
2. Append the Integration Key
3. UTF8 encode the string, note in PHP this should not be necessary, depending on your configuration.
4. Create a SHA512 hash of the string and output the result as upper case hexadecimal.

This value should match the hash value in the post, if not do not accept the notification.

URL Safe Base 64 Encoding

Base 64 encoding of strings can result in characters like +, = and / which are **not** URL safe i.e. if included in a URL they would change the meaning/path of the URL.

For this reason Paynow uses **URL encoded, base 64 encoded** strings for some arguments. For your convenience, these methods are listed below

N.B. You can also do this online at:

Base 64 Encode - <http://www.freeformatter.com/base64-encoder.html>

URL Encode - <http://www.freeformatter.com/url-encoder.html>

C#

```
private string GenerateLink(string reference, int locked, decimal amount, string merchantEmail, string customerEmail)
{
    const string paynowUrl = "https://www.paynow.co.zw";

    string arguments = string.Format(
        "search={0}&amount={1}&reference={2}&l={3}",
        HttpUtility.UrlEncode(merchantEmail),
        HttpUtility.UrlEncode(amount.ToString("F02")),
        HttpUtility.UrlEncode(reference),
        locked);

    arguments =
    HttpUtility.UrlEncode(Convert.ToBase64String(Encoding.ASCII.GetBytes(arguments)));

    return string.Format("{0}/payment/link/{1}?q={2}", paynowUrl, customerEmail, arguments);
}
```