

TXTNATION

JunglePay API

Complete XML Feed

txtNation Development
Revision (September 2010)

Connect your system to a complete interactive billing platform. A completely seamless integration using XML and direct SMS / IVR billing connectivity.

JunglePay API

Welcome to the JunglePay API documentation. In this documentation you can find the following information:

- Introduction
- Feed URL Location
- Feed Parameters
- Dynamic Tools
 - Dynamic Payments
 - Pricing Monitor
- XML Fields Explained
 - Status
 - Data
 - Flow
 - Languages
 - Countries
- Using XML Data
- Checking Payment Progress
- End-User Payment Flow
 - Premium SMS
 - Premium Phone
 - Credit Card
- Password Validation
- Example Data
- Example Integration (PHP)
- Payment Notifications (Remote Assist)

Introduction

The JunglePay API allows you to connect to JunglePay's main core and return a feed of XML data that will assist you in building your very own interface and functionality to JunglePay. The XML feed is broadcast in 19 different languages and the request of a single parameter. All SMS and IVR countries that we support on the regular JunglePay widget are also available on the XML feed. This guide will help you through using the JunglePay API and its many features.

The API can be used to develop an interactive payment gateway on many different platforms. Including flash. Standard XML can be parsed easily in most languages.

Feed URL Location

You will find the XML feed located within the same area as JunglePay, it is a HTTP link and all parameters are passed as GET vars.

HTTP URL	http://menable.txtnation.com/JunglePay/api/	<i>The URL of the XML feed</i>
-----------------	---	--------------------------------

Feed Parameters

The main parameters required to get started can be found below with an example of the data and explanation as to what they represent.

wkey	2a8f07012f2b08ea1785906f8a7aa4d0dbf85df1	<i>Key identifying your campaign from CP.</i>
referer	test	<i>Unique identifier to identify the transaction in your system (optional)</i>
language	EN	<i>The language code of the end user</i>
country	UK	<i>The country of the end user</i>
trans_id	ER4384	<i>The existing transaction code if exists (stage dependant)</i>
ip	192.168.0.1	<i>The IP address of the end user (optional)</i>
password	F38HXW	<i>Send the password for validation (when required)</i>

Dynamic Tools

Dynamic Payment's will need to send an additional parameter in all cases. This parameter will contain a secure code that will relate to the individual payment only. If you are using the 'secure' parameter you will not need to use the 'trans_code' parameter at all.

secure	SE44MDI2NDA,	The code used to make a dynamic price point payment.
---------------	--------------	--

Generating a dynamic payment you will need to refer to the Dynamic Tools documentation that can be found in Control Panel 5.1 documentation section.

Pricing Monitor is also a tool that you may require when using the dynamic payment system or even a static payment. You can send any price point in any currency to the small API and it will return you the exact value that the end user will have to pay, listing all 3 of the payment methods i.e. **Premium SMS**, **Premium Phone** and **Credit Card**. The documentation for this can also be found in the Control Panel 5.1 documentation section under Dynamic Tools.

XML Fields Explained

All the fields in the XML are explained as to their purpose below:

Block '**STATUS**'

code	1	The status of the current transaction
reference	setp_1	The current payment step that the end user is on
info	Waiting for payment to be started	A short explanation of the transaction status

The status of the transaction has the ability to change every time the script is recalled with a transaction ID already set. This is to enable you to keep track of your user's progress with their payment and make the overall experience more interactive.

Block '**DATA**'

wkey	2a8f07012f2b08ea1785906f8a7aa4d0dbf85df1	The key for your campaign
type	sms	The type of payment to be made
country	UK	The country for the current payment method
language	EN	The language the payment data is displayed
cost	1.00	Cost of the transaction to be made
currency	GBP	Currency of the transaction to be made
pricing	£ 1.00	A formatted version of the currency
message	PAY QR461161	The message to be sent in to make the payment
shortcode	60999	The number to which the message should be sent
trans_id	QR461161	The ID of the transaction (Return to check status)
is_pin	0	Will be set to 1 if the payment is using a pin system
is_recurring	0	Will be set to 1 if the payment is recurring

All the data that is related to the current campaign included price, selected language, country, short codes and if the campaign is a recurring payment or requires the pin system. Please do not cache this data as it changes per transaction and differs per country / language select.

Block '**FLOW**'

step_1	Text PAY IY775260 to 60999	Tell the user how to start their payment
step_2	Wait for message containing password.	Tell the user to wait for their password message
step_3	Enter your password:	Tell the user to enter their password here
step_4_complete	Payment OK	Text to say the payment has been complete
step_4_error	Payment Error	Text to say that the password was incorrect
info_pin_title	Multiple Pin Payment	Pin system in use, display id required (TITLE)
info_pin_text	You are required to text in multiple	Pin system in use, display id required (INFO)
info_rec_title	Recurring Payment	Recurring Payment, display if required (TITLE)
info_rec_text	This is a subscription service ...	Recurring Payment, display if required (INFO)
info_rec_price	Price: £ 1.00	Recurring Payment, display if required (PRICE)
info_rec_freq		Recurring Payment, display if required (FREQUENCY)
info_terms_title	Terms of Service	Title for the terms text
info_terms_text	This is NOT a subscription service. It will ...	Terms end user will be bound to by using service

All the instructions the user will require to make a payment, all the text provided in the above block is fully translated in all the available languages that the JunglePay API provides. Please be sure that you do not cache any of this data as it changes often and per transaction.

Block '**LANGUAGES**' – Each contained in a block '**LANGUAGE**'

code	EN	Available language code for this service
name	English	The language name for the above code

All languages in JunglePay are available in this list, make sure you do not cache this data as languages can be updated and added all the time.

Block '**COUNTRIES**' – Each contained in a block '**COUNTRY**'

code	UK	Available country code for this service
name	United Kingdom	The country name for the above code

All supported countries for the type of service and price point that you are offering. Make sure this list is not cached as it is updated and added to all the time

Using XML Data

Using the XML data to create to lists for the end users to select their country and language is very simple. To create a list of languages that the service is available for, you will need to take each language inside the blocks '**language**' one by one and build the list by displaying the language name to the user and taking the code as the value that you can then pass back to the XML feed if selected. Again with the countries the same process is used apart from you will look for each country inside the blocks '**country**' one by one.

The front end for the user must consist of the information provided in the '**data**' block. The message for the user to the shortcode, the shortcode and the price that the user will pay as well as the currency that the price is displayed in. The transaction ID field is used to send back to the XML feed for the current payment every time you want to check the status or confirm any information.

To make use of the language translations and to help you integrate and build your service you may want to use the predefined steps with the translated text.

Checking Payment Progress

To check on the progress of your payment you will need to add the '**trans_id**' to your parameters before recalling the XML feed. You will be returned with the full feed again with any necessary changes made. The main thing you will need to monitor is the '**status**' block which will changes the transaction progresses. All possible stats can be found below:

1	Waiting for payment to be started
2	Waiting for user to receive password message
3	Waiting for user to confirm password
4	Password confirmed successfully
5	Password validation error
6	Wait for user to receive and enter pin code (SMS ONLY)
7	Repeat step 1 until pin codes are complete, then step 2 (SMS ONLY)

Status 6 is when the pin system is used to build up the payment total, once you have text in and entered enough pin codes you will then be able to proceed to status '2','3' and then either end up with '4' or '5'.

End-User Payment Flow

The end-user payment flow is explained step by step so you can get an idea as to how the service should run once complete.

- **Premium SMS** – *The user will load JunglePay and read the on screen instructions, Step 1 requires the user to send the message on screen via text message (SMS) to the shortcode provided. The user will then have to wait for a message containing a password (Step 2). Next the user will need to enter their received password into a password entry box that can be provided in Step 3 or referred to in Step 3.*
- **Premium Phone** – *The user will load JunglePay and read the on screen instructions, Step 1 requires the user to make a phone call from a landline or supported mobile network to the number provided. Once the call is initiated the user will be prompted to enter their order number found in step 1 followed by the pound or hash key on their phones keypad. The phone will then confirm the price and the time left on the call. Once complete the call will hang up and Step 2 will then display the password to the user. The user will now need to enter the password into a password entry box that can be provided in Step 3 or referred to in Step 3.*
- **Credit Card** – *The user will load JunglePay and read the on screen instructions, Step 1 requires the user to enter their email address in provided input box. In Step 2 the user will click the link to that will take them to the credit card provider which will then process their payment. Once complete the call will hang up and Step 2 will then display the password to the user. The user will now need to enter the password into a password entry box that can be provided in Step 3 or referred to in Step 3.*

Password Validation

Once the user has completed the first part of the payment they will receive a password to confirm that the full bill has been processed. The user will then need to enter this password

and have it validated so that the payment will have a status of complete and all relevant payment notifications can be sent from txtNation.

To validate the password you must use the extra post parameter '**password**' mentioned in the *Feed Parameters* section as '**when required**' the status of the transaction will then change accordingly in the results. If the password is valid the results will be '**4**' and if the password is invalid then the status will be '**5**'.

Example DATA

Below you can find the example of a URL being called and the data that has been returned in XML format. This is the layout that you will receive back. Countries and Languages have been chopped out of this data just for the example.

<http://menable.txtnation.com/widget/api/?wkey=2a8f07012f2b08ea1785906f8a7aa4d0dbf85df1&country=UK&language=EN>

```
<JunglePay_api title="mENABLE JunglePay API" version="1.0.0">
<status desc="The status of the current transaction">
  <code>1</code>
  <reference>step_1</reference>
  <info>Waiting for payment to be started</info>
</status>
<data desc="The data required to build your own JunglePay">
  <wkey>2a8f07012f2b08ea1785906f8a7aa4d0dbf85df1</wkey>
  <type>sms</type>
  <country>UK</country>
  <language>EN</language>
  <cost>1.00</cost>
  <currency>GBP</currency>
  <pricing>£ 1.00</pricing>
  <message>PAY NI599030</message>
  <shortcode>60999</shortcode>
  <trans_id>NI599030</trans_id>
  <is_pin>0</is_pin>
  <is_recurring>0</is_recurring>
</data>
<flow desc="The flow that the end-user should follow">
  <step_1>Text PAY NI599030 to 60999</step_1>
  <step_2>Wait for message containing password.</step_2>
  <step_3>Enter your password:</step_3>
  <step_4_complete>Payment OK</step_4_complete>
  <step_4_error>Payment Error</step_4_error>
  <info_pin_title>Multiple Pin Payment</info_pin_title>
  <info_pin_text>You are required to text in multiple times to complete your payment. You will receive a number of PIN's via SMS,
which you will need to enter into this box, before clicking 'Redeem Pin'. Once you have entered all of your PIN's correctly, mENABLE will
inform you. You will then be able to access your content by using the final password received.</info_pin_text>
  <info_rec_title>Recurring Payment</info_rec_title>
  <info_rec_text>This is a subscription service payment, you will be billed periodically, to the schedule shown below, once you have
subscribed.</info_rec_text>
  <info_rec_price>Price: £ 1.00</info_rec_price>
  <info_rec_scale>Rebill: </info_rec_scale>
  <info_terms_title>Terms of Service</info_terms_title>
  <info_terms_text>This is NOT a subscription service. It will charge your phone {COST}, once.<br /><br />If you require assistance,
please feel free to utilise our Online ticketing system at: http://sd.txtnation.com, or use the number below.<br /><br />Service provided
by txtNation and powered by mENABLE.</info_terms_text>
</flow>
<languages desc="All the languages that are translated for this service type">
  <language>
    <code>BG</code>
    <name>Bulgarian</name>
  </language>
</languages>
<countries desc="All the countries that are available for this service type">
  <country>
```

```
<code>AL</code>
<name>Albania</name>
</country>
</countries>
```

Example Integration (PHP)

There is example integration available to use, located in the CP5.1 module. You will get a zip archive containing the following file list:

Folder	File	Description
/	index.php	Main file to call that will produce JunglePay integration.
Js/	prototype.js	Ajax functionality provided by a prototype extract.
Images/	tick.gif	An image used to notify the user of their progress in the payment.
Classes/	JunglePayAPI.class.php	The class that processes the feed and outputs the formatted data.
Templates/	main.tpl	The main template layout of the example JunglePay.
Templates/	instructions.tpl	The instructions only, returned by Ajax to update the screen.

All the above files can be copied to a server that supports PHP and run the index.php file in a web browser. This will allow you to see the example integration in action. You can use the XML feed directly to develop your own JunglePay widget in any language or you can build from the example JunglePay API class using PHP.

Alternative download location:

Download URL	http://app.junglepay.com/api/example_integration.zip
---------------------	---

Payment Notifications (Remote Assist)

To receive the payment notifications from the txtNation server we will send HTTP POST requests to a URL of your choice, which can be setup in Control Panel 5.1. The most secure and proven method of accepting the post requests is to use the pre-build Remote Assist system. This system manages all posts from the JunglePay system and enables txtNation support team to effectively aid end-users and password issues they may have.

Download URL	http://remoteassist.txtnation.com/releases/2.0.1/RemoteAssist-2.0.1.zip
---------------------	---

If you need further information on the remote assist system or would like to build your own please refer to the remote assist documentation found in the Control Panel 5.1 documentation section.