



# **ARTIO SMS Services**

## **HTTP API Documentation**

David Jozefov  
Michal Unzeitig

---

# **ARTIO SMS Services HTTP API Documentation**

ARTIO

Publication date: 4.9.2013

Version: 1.0.1

## **Abstract**

This document describes the usage of ARTIO SMS Services HTTP API.

## **Table of Contents**

1. Introduction .....	1
2. Getting Started .....	2
3. Usage .....	3
3.1. Basic Request Format .....	3
3.2. Supported Commands .....	3
3.3. List of Error Codes .....	6
A. SMS Messages Text Specification .....	7

# Chapter 1. Introduction

ARTIO SMS Services HTTP API allows developers to access functionality of ARTIO SMS Services programmatically using simple HTTP GET or POST requests. This document describes prerequisites needed to access the API and its usage.

# Chapter 2. Getting Started

Before you can use ARTIO SMS Services API, you first need to have a valid user account registered on ARTIO website. Then you need to log in to our website and activate ARTIO SMS Services. After the activation you will be assigned a unique API key which you can use along with your ARTIO username to access SMS Services API.

# Chapter 3. Usage

ARTIO SMS Services HTTP API is accessed using the HTTP protocol which makes it independant on the programming language you use. We will use PHP with cURL extension in our examples below, but generally any language that has the ability to connect to remote server using TCP can be used. We will also use POST HTTP requests in our examples, but GET requests can be used as well.

## 3.1. Basic Request Format

You can send commands to or request information from our API by sending HTTP requests to specific URL with some parameters. The main URL to contact our API is:

```
http://www.artio.net/index.php?option=com_artiosms&controller=api
```

Every command requires you to set at least the following three parameters:

task	The command to be processed by our API. See the next section 3.2 Supported Commands for the list of available commands with their parameters.
username	Your ARTIO account username.
api_key	Your ARTIO SMS Services account API key. You can find your API key in your ARTIO SMS Services Dashboard when you log in to ARTIO website.

All parameters you send should be URL encoded. If you build and send your request correctly, our HTTP API will respond with a JSON encoded message. See JSON website or Wikipedia article for more information about JSON format syntax. The response message will be an object containing at least the following two properties:

success	(boolean) - true if your request succeeded or false if there was some error encountered during processing your request.
err	(integer) - numerical code of the error encountered. 0 if the request succeeded, positive number if failed. See section 3.3 List of Error Codes for the list of possible values and their meaning.

In addition, the response object will also contain the `msg` property if your request failed, with a human readable message describing the error encountered. The response object can also contain properties specific to requested command with some information - for example when you request the current credit amount available for your account.

## 3.2. Supported Commands

This section lists all the commands supported by ARTIO SMS Services HTTP API along with their parameters and response data.

### 3.2.1. get\_credit\_info

Returns information about currently available credit amount for your account.

#### Command parameters

(none)

**Response properties**

- credit (float) - floating point value of currently available credit for your account.
- exhaustion (integer) - numerical value of estimated time to your credit exhaustion in hours. This value is computed from the last 14 days activity of your account.

**Example code**

```
// Build POST request data string
$post = 'task=get_credit_info&username=xxxxxx&api_key=xxxxxx';

// Send request
$ch = curl_init('http://www.artio.net/index.php?option=com_artiosms&controller=api');
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $post);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$out = curl_exec($ch);
curl_close($ch);

// Decode JSON response into object
$response = json_decode($out);
```

### **3.2.2. send\_sms**

Sends SMS message from your account to specified phone number.

**Command parameters**

- to Recipient's phone number in international format (with country calling code) without leading plus symbol or zeroes. All non-numerical characters will be ignored.
- text Text of your SMS message. There are certain limitations for the characters you can use that you must bear in mind. See SMS Messages Text Specification appendix for more information.
- allowUnicode Whether this SMS can be sent as Unicode if required. Allowed values for this parameter are 1 (allow) and 0 (don't allow). See SMS Messages Text Specification appendix for more information about Unicode SMS messages.
- src [optional] Application that was used to send the SMS. This information is used only for statistics and debugging purposes and is not required.

**Response properties**

(none)

**Example code**

```
// Build POST request data string
$post = 'task=send_sms&username=xxxxxx&api_key=xxxxxx&to=420123456789&text=This%20is%20a%20testing
%20SMS&allowUnicode=1';

// Send request
$ch = curl_init('http://www.artio.net/index.php?option=com_artiosms&controller=api');
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $post);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$out = curl_exec($ch);
```

```

curl_close($ch);

// Decode JSON response into object
$response = json_decode($out);

```

### 3.2.3. send\_own\_sms

Sends SMS message from your account to your own phone number specified in ARTIO SMS Services Configuration on ARTIO website.

#### Command parameters

text	Text of your SMS message. There are certain limitations for the characters you can use that you must bear in mind. See SMS Messages Text Specification appendix for more information.
allowUnicode	Whether this SMS can be sent as Unicode if required. Allowed values for this parameter are 1 (allow) and 0 (don't allow). See SMS Messages Text Specification appendix for more information about Unicode SMS messages.
src	[optional] Application that was used to send the SMS. This information is used only for statistics and debugging purposes and is not required.

#### Response properties

(none)

#### Example code

```

// Build POST request data string
$post = 'task=send_own_sms&username=xxxxxx&api_key=xxxxxx&text=This%20is%20a%20testing
%20SMS&allowUnicode=1';

// Send request
$ch = curl_init('http://www.artio.net/index.php?option=com_artiosms&controller=api');
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $post);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$out = curl_exec($ch);
curl_close($ch);

// Decode JSON response into object
$response = json_decode($out);

```

### 3.2.4. get\_logs

Returns last 50 logs stored for your account.

#### Command parameters

(none)

#### Response properties

data	(array) - array of individual logs. Each log has the following properties:
	<ul style="list-style-type: none"> <li>• <i>timestamp</i> - the Unix timestamp in UTC time</li> </ul>
	<ul style="list-style-type: none"> <li>• <i>action</i> - one letter code for specific action: S (SMS sent), E (SMS sending error), C (credit charge)</li> </ul>

- *number* - SMS recipient phone number
- *credit* - credit change, positive for credit charge action, negative for SMS sent action
- *desc* - detailed error message for SMS sending error

#### Example code

```
// Build POST request data string
$post = 'task=get_logs&username=xxxxxx&api_key=xxxxxx';

// Send request
$ch = curl_init('http://www.artio.net/index.php?option=com_artiosms&controller=api');
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $post);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$out = curl_exec($ch);
curl_close($ch);

// Decode JSON response into object
$response = json_decode($out);
```

### 3.3. List of Error Codes

The table below lists all the possible error codes returned from ARTIO SMS Services HTTP API along with their description.

**Table 3.1. Possible error codes**

Error code	Description
1	The username and API key you provided in your request are not valid. Either there's no SMS Services account created for given username, or the <i>api_key</i> is incorrect.
2	Your SMS Services account is currently disabled. Log in to ARTIO website to enable it again.
3	Your SMS Services account is currently paused. Log in to ARTIO website to resume it again.
4	Some parameters supplied for the specified command are missing or invalid. Check if you specified all the required parameters correctly.
5	SMS could not be sent. Either there's not enough credit left on your account, the recipient's phone number is invalid or there's some problem with the SMS gateway.

# Appendix A. SMS Messages Text Specification

Due to inner implementation of SMS messaging protocol, there are some limitations on the characters you can use and the length of the messages.

Each SMS message can be transferred either as standard SMS or Unicode SMS. Standard SMS can contain following standard characters (SP stands for space) and extended characters:

**Table A.1. Standard SMS characters**

SP!	"	#	%	&	'	(	)	*	+	,	-	.	/	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	
?	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	a	b	c
d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	@	\$	è	é	ù	ì	ò
£	¥	¤	Ø	ø	Ç	Å	å	Δ	Φ	Γ	Λ	Ω	Π	Ψ	Σ	Θ	Ξ	Æ	æ	ß	É	í	¿	Ä	Ö	Ñ	Ü	ä	ö
ñ	ü	à	§	_																									

**Table A.2. Extended SMS characters**

^	{	}	\	[	]	~		€
---	---	---	---	---	---	---	--	---

One standard SMS message can contain up to 160 characters, but extended characters are counted as two characters each.

If you need to use other characters in your SMS messages, the only other option is to send them as Unicode. Such SMS can contain any unicode characters (which is pretty much any character), but their length is limited to 70 characters only.

Furthermore, you can also send longer messages split to multiple SMS. However, you are still limited to 3 messages length only and this feature decreases one message length by 7 characters. So split standard SMS can contain 153 characters each and split Unicode SMS only 63 characters each. This gives you a maximum length limit of 459 characters for standard SMS and 189 characters for Unicode SMS. Please note that each SMS will be charged separately, so a message split to 3 SMS will cost three times the price of a single SMS.

If you are further interested in SMS protocol's inner workings, you can check this article on Wikipedia: [http://en.wikipedia.org/wiki/GSM\\_03.38](http://en.wikipedia.org/wiki/GSM_03.38).