

# text2reach SMS API

Tuesday, 18 January 2011

# v1.1

---

This document describes application interface (API) between SMS service provider (SP) and text2reach SMS gateway (SMSGW). SMSGW supports mobile originated (MO) and mobile terminated (MT) Premium SMS (paid services) and Bulk SMS.

## Contents

Contents .....	2
Premium SMS HTTP interface .....	3
Mobile Originated (MO) messages .....	3
GET parameters .....	3
XML response .....	3
List of available prices .....	4
Delivery and billing reports .....	5
Mobile Terminated (MT) messages (LMT, Tele2) .....	6
GET parameters .....	6
Response codes .....	7
PIN messages .....	7
Bulk SMS HTTP interface .....	8
GET parameters .....	8
Response codes .....	9
Delivery reports .....	9
ANNEX .....	10
GSM 7bit alphabet .....	10



Bulk SMS may be sent worldwide (testing may be required). This document covers Premium SMS services for Latvian operators only – “Latvijas Mobilais Telefons” (LMT), “Tele2 Latvia”, “Bite Latvia”. For availability in different countries and pricing, please contact our customer support!

## Premium SMS HTTP interface

### Mobile Originated (MO) messages

Every MO message sent to shared number (destination numbers are shared between multiple SPs) and starting with keyword registered to you is then sent to your server using HTTP GET request. If there is no correct response in 10s or address is not accessible, SMSGW retries this request with increasing intervals 11 times (1min, 4min, 9min, 16min, 25min, 36min, 49min, 64min, 81min, 100min, 121min).

#### Example:

If your address is <http://yourpage.com/sms/>, shared number is 1800 and your registered keyword is ‘KEYWORD’ then if SMSGW receives message containing text “KEYWORD test message” addressed to 1800, it will make the following HTTP request:

[http://yourpage.com/sms/?phone=37120034375&message=KEYWORD%20test%20message&code=1800&operator=T&msg\\_id=12345678&keyword=KEYWORD](http://yourpage.com/sms/?phone=37120034375&message=KEYWORD%20test%20message&code=1800&operator=T&msg_id=12345678&keyword=KEYWORD)

#### GET parameters

Key	Value	Description
phone	11 decimal characters	User number- source address in international form.
code	3-4 decimal characters	Shared number- destination address.
operator	1,2 character	L = LMT, TP = Tele2 Latvia, TZ – Zelta Zivtiņa, B = Bite Latvia.
message	Up to 160 characters	Message text or binary content.
keyword	Up to 50 characters	Keyword is used to recognize SP (destination number is shared between multiple SPs).
msg_id	32bit unsigned integer	Unique MO id number. In case of repeated request caused by error, this number does not change.

#### XML response

In response to this request server must respond with XML content like this:

```
<?xml version="1.0" encoding="utf-8"?>
<sms>
  <price>75</price>
  <text>This is your response!</text>
</sms>
```

Where:

- **price** – end user price in 0.01 LVL;
- **text** – response text up to 160 characters (see [GSM 7bit alphabet](#) in ANNEX);



Latvian special characters will be replaced with their Latin counterpart ('ā'-'a', 'ņ'-'n' etc.); all other characters outside [GSM 7bit alphabet](#) will be removed.

If you want to respond with binary content then response should look like this:

```
<?xml version="1.0" encoding="utf-8"?>
<sms>
  <price>75</price>
  <binary>
0605040B8423F025060803AE81EAAF82B48401056A0045C60C036B696E67736D732E636F6D00080103574
150205075736820757A204B696E67534D53206C617075000101
  </binary>
</sms>
```

Where:

- **price** – end user price in 0.01 LVL;
- **binary** – HEX encoded response data;

If you want to send text message which contains characters not available in [GSM 7bit alphabet](#) e.g. "ā", "č", "ж", you should respond with Unicode message:

```
<?xml version="1.0" encoding="utf-8"?>
<sms>
  <price>75</price>
  <unicode>ЭТО ВАШ ОТВЕТ!</unicode>
</sms>
```

Where:

- **price** – end user price in 0.01 LVL;
- **unicode** – response text up to 70 characters;

If you do not wish to respond then you should respond with:

```
<?xml version="1.0" encoding="utf-8"?>
<sms>
  <noresponse/>
</sms>
```

## List of available prices

Price in 0.01 LVL	End user price
0*	0.00 LVL
7	0.07 LVL
10	0.10 LVL
15	0.15 LVL
20	0.20 LVL
25	0.25 LVL



30	0.30 LVL
35	0.35 LVL
40	0.40 LVL
45	0.45 LVL
50	0.50 LVL
55	0.55 LVL
60	0.60 LVL
75	0.75 LVL
85	0.85 LVL
95	0.95 LVL
100	1.00 LVL
125	1.25 LVL
150	1.50 LVL
175	1.75 LVL
200	2.00 LVL
225	2.25 LVL
250	2.50 LVL
275	2.75 LVL
300	3.00 LVL

\* – must be enabled in SMSGW.

In addition to end user price set by SP- LMT customers must pay standard SMS price (depends on contract between operator and end user).

Billing is done using response message. If billing fails, user will not receive that message (LMT customers will receive info message if there is insufficient balance in prepaid account). If end user is billed for other services than message text itself, SP should check billing status before providing those services.

### Delivery and billing reports

SMSGW supports delivery and billing reports. SMSGW makes HTTP GET request to address provided by SP with following parameters:

- **msg\_id** – MO id of message this report belongs to;
- **status** – status of response message;
  - charged – billing successful;
  - notcharged – billing failed;
  - delivered – response message delivered;
  - undelivered – response message was not delivered.
- **extra** – optional parameter that describes the reason of billing/delivery failure (see table below).



In response to this HTTP request server must respond with acknowledgement 'OK'. If there is no correct response in 10s or address is not accessible, SMSGW retries this request with increasing intervals 11 times (1min, 4min, 9min, 16min, 25min, 36min, 49min, 64min, 81min, 100min, 121min).

Extra	Description
0	Success.
1	Retry count reached maximum.
2	Final status.
3	Number does not exist. Do not resend the message! Please, remove this number from any active database and do not make any transactions without new MO from this number.
4	Low balance. Insufficient funds in end user prepaid account.
5	Premium services are disabled for this end user.
6	End user is out of reach.
7	Wrong sender address.
8	Rejected by spam filter.
9	SMS content corrupted.
10	Temporary system error.
11	Price does not exist.
13	SMS too long.
14	Wrong UDH (User Data Header).
15	Wrong data coding.
17	Number temporarily disabled.
18	Monthly billing limit reached.
19	Billing cancelled by user.
20	Unspecified error.
21	Operator integration error.
22	Expired at SMSGW.
23	Destination operator not allowed.
24	Not enough funds in service providers account.

## Mobile Terminated (MT) messages (LMT, Tele2)

SMSGW provides ability to send one or more billing messages asynchronously in response to one single MO. This is very useful for subscription services or services where response is not immediately available.

### GET parameters

To send MT SP must request the following URL:

[https://gw.kingsms.com/send/sms/?api\\_key=API\\_KEY&phone=PHONE&message=TEXT&type=txt&price=PRICE&inboxid=MOID](https://gw.kingsms.com/send/sms/?api_key=API_KEY&phone=PHONE&message=TEXT&type=txt&price=PRICE&inboxid=MOID)

Where:

- **api\_key** – your registered MT API key;



- **phone** – end user phone number in international form;
- **message** – text up to 160 characters (see [GSM 7bit alphabet](#) in ANNEX);
- **type** – message type ('txt' for text, 'bin' for binary, 'pin' for PIN codes);
- **price** – see [List of available prices](#);
- **inboxid** – MO id of the original message or PIN id number.\*

\* – These numbers serve as reason for the transaction (e.g. subscription or delayed response to original MO message).

If type is 'txt', Latvian special characters will be replaced with their Latin counterparts ('ā'→'a', 'ņ'→'n' etc.) and all other characters outside [GSM 7bit alphabet](#) will be removed. Messages of type 'bin' must be HEX encoded.

## Response codes

In response you will receive unique positive integer (MT message id) or negative integer in case of error.

Response	Description
> 0	Success. MT message id.
-400	Wrong API key <b>api_key</b> .
-500	Missing parameters.
-501	Wrong <b>type</b> (must be 'txt' or 'bin').
-502	Wrong or missing <b>price</b> .
-503	Destination address blocked.
-504	Not available for this operator.
-505	<b>price</b> exceeds maximum.
-507	<b>price</b> does not exist.
-508	Wrong destination address.
-509	Wrong message encoding.
-510	Wrong <b>inboxid</b> .
-511	Number does not exist or operator/owner has been changed.
-513	Wrong message length.
-555	System error.

Also see [List of available prices](#) and [Delivery and billing reports](#).

## PIN messages

'pin' message is special type of text message that allows you to start subscription without MO message. Instead of MO message PIN code is used to start subscription like this:

1. End user enters its phone number in web page of SP.
2. SP sends MT message of type 'pin' (parameter **inboxid** omitted) with text like: "Your PIN code is '@@!'" **Note** that maximum of 140 characters can be used.



3. SMSGW replaces @@ with generated PIN code and sends it to end user.
4. End user enters this PIN code in web page of SP.
5. SP verifies PIN code by making HTTP request  
[https://gw.kingsms.com/send/pin/?api\\_key=API\\_KEY&phone=PHONE&pin=PIN](https://gw.kingsms.com/send/pin/?api_key=API_KEY&phone=PHONE&pin=PIN). In response SP receives PIN id or error code.
6. SP uses PIN id as **inboxid** parameter to send MT messages.

Response	Description
>0	Success. PIN id.
-100	Wrong API key <b>api_key</b> .
-101	Missing phone number.
-102	Missing PIN id.
-103	Wrong PIN id.
-110	Internal system error.

For set of rules subscription services must comply, please see “Terms and Conditions of Text2reach MT billing subscription service”.

## Bulk SMS HTTP interface

### GET parameters

To send a message SP must make HTTP GET request to address <https://gw.kingsms.com/send/bulk/v2/> with parameters described below:

- **api\_key** – your registered Bulk API key;
- **phone** – destination address in international form;
- **message** – plain text message up to 459\* characters, Unicode message up to 201\* characters, binary message up to 140 bytes.
- **type** – message type ('txt' for text, 'bin' for binary);
- **ucs2** – true or false, if set to true- message text will be encoded in Unicode if necessary, otherwise Latvian special characters will be replaced with their Latin counterparts ('ā' → 'a', 'ņ' → 'n' etc.) and all other characters outside [GSM 7bit alphabet](#) will be removed.
- **timestamp** – (optional) Unix timestamp of a time when end user opted in to receive messages. If parameter is not set the current timestamp is used\*\*.
- **report\_url** – (optional) URL for delivery reports (e.g. <http://yourpage.com/bulk/reports.php>).
- **expires** – (optional) Unix timestamp of a time when to stop trying to deliver message. If message is not delivered before time specified you will receive 'expired' status. Exact behaviour depends on network operators (different, operators may have different maximum and default validity times and this parameter may even be ignored)!

\* – Longer messages are achieved using multipart concatenated messages (up to 3 parts). If plain text message is longer than 160 characters, message is split across multiple parts up to 153 characters each (Unicode messages 70 and 67 accordingly). **Note** that you will be credited for each part.



\*\* – If owner of a phone number is changed, number is closed or operator is changed, message will not be delivered. If new owner of a number opts in to receive data, new timestamp should be used.

**Example:**

[https://gw.kingsms.com/send/bulk/v2/?api\\_key=API\\_KEY&ucs2=true&type=txt&phone=37120034375&message=Hello!&timestamp=1271421881&expires=1271425481&report\\_url=http://yourpage.com/bulk/reports.php](https://gw.kingsms.com/send/bulk/v2/?api_key=API_KEY&ucs2=true&type=txt&phone=37120034375&message=Hello!&timestamp=1271421881&expires=1271425481&report_url=http://yourpage.com/bulk/reports.php)

Response is positive integer if SMSGW accepted message for delivery. If response is 0 or negative- error occurred.

**Response codes**

Response	Description
0	Spam filter. One or more identical messages in 5 minute timeframe.
> 0	Success. Unique message id number. Used in reports to identify sent message.
-400	Wrong API key <b>api_key</b> .
-410	Not enough funds in your account.
-411	Wrong <b>expires</b> value.
-500	Missing parameters.
-503	Destination address blocked.
-508	Wrong destination address.
-509	Wrong message encoding.
-511	Number does not exist or operator/owner has been changed.
-513	Wrong message length.
-555	System error.
-556	Test successful.

**Delivery reports**

At the moment SMSGW receives delivery report- HTTP request is made to address set in **report\_url** with following parameters:

**msg\_id** – message id number returned when SMS was sent.

**status** – delivery status

- delivered – SMS is delivered.
- undelivered – SMS delivery failed.
- expired – SMS could not be sent during validity time.

In response to this HTTP request server must respond with “200 OK” HTTP header. If there is no correct response in 10s or address is not accessible, SMSGW retries this request with increasing intervals 11 times (1min, 4min, 9min, 16min, 25min, 36min, 49min, 64min, 81min, 100min, 121min).



## ANNEX

### GSM 7bit alphabet

Character	Unicode Hex	UTF-8 Hex	Character	Unicode Hex	UTF-8 Hex	Character	Unicode Hex	UTF-8 Hex	
@	0040	40	%	0025	25	S	0053	53	
£	00A3	C2A3	&	0026	26	T	0054	54	
\$	0024	24	'	0027	27	U	0055	55	
¥	00A5	C2A5	(	0028	28	V	0056	56	
è	00E8	C3A8	)	0029	29	W	0057	57	
é	00E9	C3A9	*	002A	2A	X	0058	58	
ù	00F9	C3B9	+	002B	2B	Y	0059	59	
ì	00EC	C3AC	,	002C	2C	Z	005A	5A	
ò	00F2	C3B2	-	002D	2D	Ä	00C4	C384	
Ç	00C7	C387	.	002E	2E	Ö	00D6	C396	
LF	000A	0A	/	002F	2F	Ñ	00D1	C391	
∅	00D8	C398	0	0030	30	Ü	00DC	C39C	
ø	00F8	C3B8	1	0031	31	Ş	00A7	C2A7	
CR	000D	0D	2	0032	32	ı	00BF	C2BF	
Å	00C5	C385	3	0033	33	a	0061	61	
å	00E5	C3A5	4	0034	34	b	0062	62	
Δ	0394	CE94	5	0035	35	c	0063	63	
_	005F	5F	6	0036	36	d	0064	64	
Φ	03A6	CEA6	7	0037	37	e	0065	65	
Γ	0393	CE93	8	0038	38	f	0066	66	
Λ	039B	CE9B	9	0039	39	g	0067	67	
Ω	03A9	CEA9	:	003A	3A	h	0068	68	
Π	03A0	CEA0	;	003B	3B	i	0069	69	
Ψ	03A8	CEA8	<	003C	3C	j	006A	6A	
Σ	03A3	CEA3	=	003D	3D	k	006B	6B	
Θ	0398	CE98	>	003E	3E	l	006C	6C	
Ξ	039E	CE9E	?	003F	3F	m	006D	6D	
FF	000C	0C	<	i	00A1	C2A1	n	006E	6E
^	005E	5E	<	A	0041	41	o	006F	6F
{	007B	7B	<	B	0042	42	p	0070	70
}	007D	7D	<	C	0043	43	q	0071	71
\	005C	5C	<	D	0044	44	r	0072	72
[	005B	5B	<	E	0045	45	s	0073	73
~	007E	7E	<	F	0046	46	t	0074	74
]	005D	5D	<	G	0047	47	u	0075	75
	007C	7C	<	H	0048	48	v	0076	76
€	20AC	E282AC	<	I	0049	49	w	0077	77
Æ	00C6	C386	J	004A	4A	x	0078	78	
æ	00E6	C3A6	K	004B	4B	y	0079	79	
ß	00DF	C39F	L	004C	4C	z	007A	7A	
É	00C9	C389	M	004D	4D	ä	00E4	C3A4	
SP	0020	20	N	004E	4E	ö	00F6	C3B6	
!	0021	21	O	004F	4F	ñ	00F1	C3B1	
"	0022	22	P	0050	50	ü	00FC	C3BC	
#	0023	23	Q	0051	51	à	00E0	C3A0	
¤	00A4	C2A4	R	0052	52				

Single plain text SMS may contain up to 160 7bit septets. All characters occupy 1 septet except those marked with '<' occupy two septets. Keep this in mind when calculating length of message!

