

Bulk SMS API Documentation (Version 4.5)

CONTENTS

1) Introduction	2
2) API Version & Updated Details	2
3) Major Updates in 4.5 Version	2
4) General Usages of API's	3
5) How API Works	3
6) API for Sending Single SMS (Normal Route)	5
7) API for Sending Single SMS (Transaction Route)	5
8) API for Sending Single SMS (Priority Route)	6
9) API for Sending Single SMS (Marketing Route)	6
10) API for Sending Multiple SMS (Normal Route)	6
11) API for Sending Multiple SMS (Priority Route)	7
12) API for Sending Multiple SMS (Transactional Route)	7
13) API for Sending Multiple SMS (Marketing Route)	8
14) Triggering Transactional Messages	9
15) API for Requesting Transactional Templates	9
16) API for Dynamic Messaging using CSV Format	10
17) XML API	11
18) The variables to be used in XML Format	12
19) API for Dynamic Messaging Using XML Format	12
20) XML API for Dynamic Messaging	14
21) XML API for Bulk Push	16
22) API for Scheduling SMS	18
23) API for Deleting Scheduled SMS	18
24) API to Fetch Single Delivery Report	19
25) API to Fetch Multiple Delivery Report	20
26) API to Fetch Multiple Delivery Report using PushId	21
27) API to Fetch Datewise Delivery Report	22
28) API to Fetch Current Normal Balance	23
29) API to Fetch Current Priority Balance	23
30) API to Fetch Current Marketing Balance	23
31) API to Fetch Current Transaction Balance	23
32) API to List All Sender ID's	24
33) All Possible Responses & Errors	24
34) Description of Files Used	25

Introduction

API stands for Application Programming Integration which is widely used to integrate and enable interaction with other software, much in the same way as a user interface facilitates interaction between humans and computers. Our API codes can be easily integrated to any web or software application.

Our Robust & Stable HTTP API's is Compatible with all Programming Languages & Operating Systems. Easy to Integrate & Use

Our XML API is designed to send huge messages through a single push

API Version & Updated Details

Last Updated On : 06-April-2013
API Version : 4.5

Major Updates in 4.5 Version

- ✓ Introduced Simple Dynamic Messaging API to push thousands of Dynamic Contents through a Single API Call (CSV & XML Format)
- ✓ DND Template ID Based Matching Skipped on the Transactional Route, now instead of the exact template id you can pass "2" as fixed value i.e `tempid=2`, system will check based on the message content and if the message content is not matching with the approved content then system will give the error "DND Template Not Matching"
- ✓ Easy to use PHP Libraries to Send Messages & Receive Delivery Reports within your program/application/software
- ✓ Transactional Message Template can be request through API
- ✓ XML API for Processing Huge Messages, Dynamic XML API's also included.
- ✓ Date Wise Delivery Reports in XML/JSON/CSV Format
- ✓ Bulk Delivery Reports in XML/JSON/CSV Format based on Push ID
- ✓ Option to retrieve Push ID as API Response to fetch bulk delivery easily.

General Usages of API's

- ✓ Integrating API with your webpage enables you to send SMS from your web pages.
- ✓ Integrating API with your Desktop application enables you to send SMS from your Desktop Application
- ✓ Integrating API with your billing software enables you to send thanking messages to customers after their purchase.
- ✓ Integrating API with your payroll software enables you to send salary intimation to employees.
- ✓ Integrating API with your server enables you to send critical system warnings/errors to Server admin/client.

How API Works

The API works on GET and POST METHOD, and the url to be used is <http://yourdomain/api/sms.php>

Note : Replace yourdomain with your connecting site URL. For more details please refer the Developers API section in your control panel. Go To API & Downloads - >SMS HTTP API

For every successful execution of our API call, the system generates a 'Response' which you can save for tracking the operation.

For multiple requests, the system response will consist of unique numeric ids or other forms of responses, separated by comma.

Delimiter : We are using comma (,) as a delimiter to separate each record in a Response

UID & PIN

Our API works with Unique UID & PIN, which is not associated with your sms account username or password, even if you change your password of the sms control panel API will work, if you need to generate new PIN, please login to your control panel and Generate New PIN.

The variables to be used

uid	Your Unique Identification
pin	API authentication PIN
sender	6 Digit Alpha Message SenderID (Only for Transactional Route)
mobile	Recipient 10 Digit Mobile Number without 0 or 91
message	Message Contents to be sent, should be url encoded
route=0	For selecting Normal route
route=1	For selecting Priority route
route=4	For selecting Marketing route
route=5	For selecting Transactional Route
tempid	Pre Approved Message Template ID (Only for Transaction Route)
tempid=2	Use tempid=2 if you cant pass the exact Template Message ID, this method might delay to send messages if you have lot of pre-approved templates.
time=dd-mm-yyyy-24h-min (01-02-2011-16-38)	For Scheduling messages
unicode=1	For Sending unicode messages
flash=1	If you want delivery the message Flash then pass the value flash=1
dynamic=1	For Selecting Dynamic Mode
itype=xml/csv	In which format you are inputting the data. Supports CSV/XML Format. Applies only for Dynamic Messaging
data	Used for dynamic messages the dynamic message content need to be appended. First field should be 10 digit mobile number
rtype=xml/json/csv	How you need to get response for Delivery Reports XML/JSON/CSV
pushid=1	If you want to get the pushid of the message as response, by default you will get message id.
msgid	A numeric unique Message ID to be passed for deleting a scheduled message
pushid	A numeric unique Push ID to be passed for deleting a scheduled message

API for Sending Single SMS (Normal Route)

`http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=your_sender_id&route=0&mobile=MOBILE&message=MESSAGE&pushid=1`

sender & pushid are optional

Sample Response

223433

You will get a numeric Message ID or Push ID which you can save for tracking the message you sent.

If you pass `pushid=1` then instead of Message ID you will get Push ID as response which will be useful for bulk pushes to retrieve complete delivery status of all numbers associated to that push.

API for Sending Single SMS (Transaction Route)

`http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=SENDER&route=5&tempid=DND_TEMPLATE_ID&mobile=MOBILE1&message=MESSAGE&pushid=1`

For Transaction SMS, the sender cannot be left blank and pushid=1 is optional

Sample Response

223433

You will get a numeric Message ID or Push ID which you can save for tracking the message you sent.

If you pass `pushid=1` then instead of Message ID you will get Push ID as response which will be useful for bulk pushes to retrieve complete delivery status of all numbers associated to that push.

API for Sending Single SMS (Priority Route)

`http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=SENDER&route=1&mobile=MOBILE1&message=MESSAGE`

sender & pushid are optional

Sample Response

223433

You will get a numeric Message ID which you can save for tracking the message you sent.

If you pass `pushid=1` then instead of Message ID you will get Push ID as response which will be useful for bulk pushes to retrieve complete delivery status of all numbers associated to that push.

API for Sending Single SMS (Marketing Route)

`http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=SENDER&route=4&mobile=MOBILE1&message=MESSAGE`

sender & pushid are optional

Sample Response

223433

You will get a numeric Message ID which you can save for tracking the message you sent.

If you pass `pushid=1` then instead of Message ID you will get Push ID as response which will be useful for bulk pushes to retrieve complete delivery status of all numbers associated to that push.

API for Sending Multiple SMS (Normal Route)

`http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=your_sender_id&route=0&mobile= myrecipient1, myrecipient2, myrecipient3, myrecipient4, myrecipient5,myrecipient6&message=MESSAGE`

You can pass 10 digit mobile number as comma separated in the `mobile` field to deliver the same message content to multiple numbers.

Default Response

3287,3288,3289

Suppose you have 3 receivers and you passed the numbers in the form

mobile=9995xxxxxx, 8707xxxxxx,9020xxxxxx

If the message is sent successfully, you will get message ids corresponding to each message you sent , separated by comma.

If pushid= 1 Passed

If you pass pushid=1 then you will get only push id, with this pushid you can fetch delivery reports of all numbers in the push through a single api call

API for Sending Multiple SMS (Priority Route)

http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=your_sender_id&route=1&mobile= myrecipient1, myrecipient2, myrecipient3, myrecipient4, myrecipient5,&message=MESSAGE

You can pass 10 digit mobile number as comma separated in the **mobile** field to deliver the same message content to multiple numbers.

Default Response

3287,3288,3289

Suppose you have 3 receivers and you passed the numbers in the form

mobile=9995xxxxxx, 8707xxxxxx,9020xxxxxx

If the message is sent successfully, you will get message ids corresponding to each message you sent , separated by comma.

If pushid= 1 Passed

If you pass pushid=1 then you will get only push id, with this pushid you can fetch delivery reports of all numbers in the push through a single api call

API for Sending Multiple SMS (Transactional Route)

http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=your_sender_id&route=5&tempid=DND_TEMPLATE_ID&mobile= myrecipient1, myrecipient2, myrecipient3, myrecipient4, myrecipient5,&message=MESSAGE

For Transaction SMS, the sender cannot be left blank.

You can pass 10 digit mobile number as comma separated in the **mobile** field to deliver the same message content to multiple numbers.

Default Response

3287,3288,3289

Suppose you have 3 receivers and you passed the numbers in the form

mobile=9995xxxxxx, 8707xxxxxx,9020xxxxxx

If the message is sent successfully, you will get message ids corresponding to each message you sent , separated by comma.

If pushid=1 Passed

If you pass pushid=1 then you will get only push id, with this pushid you can fetch delivery reports of all numbers in the push through a single api call

API for Sending Multiple SMS (Marketing Route)

http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=your_sender_id&route=4&mobile= myrecipient1, myrecipient2, myrecipient3, myrecipient4, myrecipient5,&message=MESSAGE

You can pass 10 digit mobile number as comma separated in the **mobile** field to deliver the same message content to multiple numbers.

Default Response

3287,3288,3289

Suppose you have 3 receivers and you passed the numbers in the form

mobile=9995xxxxxx, 8707xxxxxx,9020xxxxxx

If the message is sent successfully, you will get message ids corresponding to each message you sent , separated by comma.

If pushid=1 Passed

If you pass pushid=1 then you will get only push id, with this pushid you can fetch delivery reports of all numbers in the push through a single api call

Triggering Transactional Messages

As our transactional gateway is based on template system, the content of the message should be pre-approved and the approved message template will be assigned with a unique template id, to get your template id please login into your panel and navigate to Messaging- >DND Templates

From 4.5 API Version onwards we have skipped the DND Template ID Based Matching on the Transactional Route, now instead of the exact template id you can pass "2" as fixed value i.e **tempid=2**, system will authenticate only based on the message content and if the message content is not matching with the pre-approved content then system will give the error "**DND Template Not Matching**"

Lets assume that your template id is **123** and the approved message content as below

Dear **#VAL#**
Your Account is credited with **#VAL#** on **#VAL#** towards **#VAL#**, your available balance is Rs **#VAL#**

Here **#VAL#** is the dynamic values where you can replace with your own dynamic values.

Example with your own values

Dear **Raj**
Your Account is credited with **Rs 2000** on **20-12-2011** towards **FD** your available balance is Rs **10,000**

If you make any changes in the format then you will get an error message "**DND Template Not Matching**"

Encode the message content , put the corresponding template id and submit the message to our Server for delivery

API for Requesting Transactional Templates

You can request any number of Transactional Message contents to be used in your transactional gateway. **All Message Contents are subjected to approval.**

http://yourdomain.com/api/template.php?uid=your_uid&pin=your_pin&message=message_contents

You will get Template ID as Response to the API Call

1323

Before submitting message please encode the message using `'urlencode'` function and for dynamic field use **#VAL#**

API for Dynamic Messaging using CSV Format

Our Simple Yet Powerful Dynamic Messaging API allows you to deliver dynamic message contents to different mobile numbers through a Single API Call

http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=your_sender_id&route=0&message=MESSAGE&dynamic=1&data=Dynamic_Contents

Now Lets take an example of a message content need to deliver to 3 mobile numbers dynamically

Sample Database Structure

#FIELD1#	#FIELD2#	#FIELD3#	#FIELD4#
9999999999	Jeevan	100	01-03-2013
8888888888	Ramu	200	02-03-2013
7777777777	Rahul	300	03-03-2013

message

Dear #FIELD2#
Please pay your child fees of #FIELD3# on or before #FIELD4#

URL Encoded Data of the above message

Dear%20%23FIELD2%23%0APlease%20pay%20your%20child%20fees%20of%20%23FIELD3%23on%20or%20before%20%23FIELD4%23

data

9999999999, Jeevan, 100, 01-03-2013 <new line>
8888888888, Ram, 200, 02-03-2013 <new line>
7777777777, Rahul, 300, 03-03-2013 <new line>

URL Encoded Data of the above message

9999999999%2C%20Jeevan%2C%20100%2C%2001-03-2013%0A8888888888%2C%20Ram%2C%20200%2C%2002-03-2013%0A7777777777%2CRahul%2C300%2C03-03-2013%0A

Please note that the first field should be 10 digit mobile number and each field should be separated by Comma (,) and each record should be separated by new line (\n)

Now just POST these values to our Server for delivery

```
http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=your_
sender_id&route=0&message=
Dear%20%23FIELD2%23%0APlease%20pay%20your%20child%20fees%20of%2
0%23FIELD3%23on%20or%20before%20%23FIELD4%23&dynamic=1&data=99
9999999%2C%20Jeevan%2C%20100%2C%2001-03-
2013%0A8888888888%2C%20Ram%2C%20200%2C%2002-03-
2013%0A7777777777%2CRahul%2C300%2C03-03-2013%0A
```

System will return you the unique message id as comma separated format which can be saved for future references

232323,2323231,2323211

If pushid=1 passed then you will get only one numeric id for all numbers

34343

XML API

The newly introduced XML APIs use data in XML format. You need to send the data in predefined XML formats using the variables given below.

The XML data should be urlencoded before passing to the API.

You may use [pushid=1](#) as optional parameter.

Sample XML data:

```
<data>
  <number>
    <field>9999999999</field> <!--FIELD1 Data-->
    <field>Jeevan </field> <!--FIELD2 Data-->
  </number>
</data>
```

The variables to be used in XML Format

<uid>your_uid</uid>	Your Unique Identification
<pin>your_pin</pin>	API authentication PIN
<sender>SENDER</sender>	6 Digit Alpha Message SenderID (Only for Transactional Route)
<number>Number</number>	Recipient 10 Digit Mobile Number without 0 or 91
<message>Message</message>	Message Contents to be sent with FIELD2, FIELD3 applies only for Dynamic XML API
<route>0</route>	For selecting Normal route
<route>1</route>	For selecting Priority route
<route>4</route>	For selecting Marketing route
<route>5</route>	For selecting Transactional Route
<unicode>1</unicode>	For Sending unicode messages, messages should be urlencoded
<flash>1</flash>	If you want to deliver the message as Flash then pass the value flash=1
<dynamic>1</dynamic>	For Selecting Dynamic Mode
<bulkpush>1</bulkpush>	For Enabling Bulk Push Mode
<pushid>1</pushid>	If you want to get the pushid of the message as response, by default you will get message id.

API for Dynamic Messaging Using XML Format

Our Simple Yet Powerful Dynamic Messaging API allows you to deliver dynamic message contents to different mobile numbers through a Single API Call.

http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=your_sender_id&route=0&message=MESSAGE&dynamic=1&itype=xml&data=dynamic_contents_in_xml_format

'itype=xml' indicates that you use xml format for dynamic contents.

Now Lets take an example of a message content need to deliver to 2 mobile numbers dynamically using xml format

Sample Database Structure

#FIELD1#	#FIELD2#	#FIELD3#	#FIELD4#
9999999999	Jeevan	100	01-03-2013
8888888888	Ramu	200	02-03-2013

message

Dear **#FIELD2#**

Please pay your child fees of **#FIELD3#** on or before **#FIELD4#**

URL Encoded Data of the above message

Dear%20%23FIELD2%23%0APlease%20pay%20your%20child%20fees%20of%20%23FIELD3%23on%20or%20before%20%23FIELD4%23

Sample XML Data

```
<data>
  <number>
    <field>9999999999</field> <!--First field should be mobile no.-->
    <field> Jeevan </field> <!--FIELD2 Data-->
    <field>100</field> <!--FIELD3 Data-->
    <field>01-03-2013</field> <!--FIELD4 Data-->
  </number>
  <number>
    <field >8888888888</field> <!--First field should be mobile no.-->
    <field> Ramu </field> <!--FIELD2 Data-->
    <field>200</field> <!--FIELD3 Data-->
    <field>02-03-2013</field> <!--FIELD4 Data-->
  </number>
</data>
```

Now you need to submit this data into our API, after urlencoding the data, as follows:

```
data = <data> <number> <field>9999999999</ field > <field> Jeevan </field>
<field>100</field> <field>01-03-2013</field> </number> <number> < field
>8888888888</ field > <field> Ramu </field> <field>200</field> <field>02-
03-2013</field> </number> </data>
```

Urlencoded form of the above data is

```
%3Cdata%3E%3Cnumber%3E%3Cfield%3E9999999999%3C/field%3E%3Cfield%
%3EJeevan%3C/field%3E%3Cfield%3E100%3C/field%3E%3Cfield%3E01-03-
2013%3C/field%3E%3C/number%3E%3Cnumber%3E%3Cfield%3E8888888888
%3C/field%3E%3Cfield%3ERamu%3C/field%3E%3Cfield%3E200%3C/field%3E
%3Cfield%3E02-03-2013%3C/field%3E%3C/number%3E%3C/data%3E
```

Now just POST these values to our Server for delivery

```
http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=your_
sender_id&route=0&message=
Dear%20%23FIELD2%23%0APlease%20pay%20your%20child%20fees%20of%2
0%23FIELD3%23on%20or%20before%20%23FIELD4%23
&dynamic=1&itype=xml&data=%3Cdata%3E%3Cnumber%3E%3Cfield%3E9999
999999%3C/field%3E%3Cfield%3EJeevan%3C/field%3E%3Cfield%3E100%3C/fi
eld%3E%3Cfield%3E01-03-
2013%3C/field%3E%3C/number%3E%3Cnumber%3E%3Cfield%3E8888888888
%3C/field%3E%3Cfield%3ERamu%3C/field%3E%3Cfield%3E200%3C/field%3E
%3Cfield%3E02-03-2013%3C/field%3E%3C/number%3E%3C/data%3E
```

System will return you the unique message id as comma separated format

```
232323,2323231
```

XML API for Dynamic Messaging

Our Simple Yet Powerful Dynamic Messaging API allows you to deliver dynamic message contents to different mobile numbers through a Single API Call.

http://yourdomain.com/api/sms.php?xml=XML_DATA

'xml=XML_DATA' indicates that you use XML API for dynamic contents.

Now Lets take an example of a message content need to deliver to 2 mobile numbers dynamically using XML API.

Sample Database Structure

#FIELD1#	#FIELD2#	#FIELD3#	#FIELD4#
9961xxxxxx	Rajeev	12	120
9400xxxxxx	Saju	15	150

message

Dear #FIELD2#,
Your account has credited with Rs #FIELD3#.Your account balance is #FIELD4#

XML Sample format

```
<data >
  <uid>your_uid</uid>
  <pin>your_pin</pin>
  <sender>SENDER</sender>
  <route>0</route>
  <dynamic>1</dynamic>
  <message>
    Dear #FIELD2#, Your account has credited with Rs #FIELD3#.Your
    account balance is #FIELD4#
  </message>

  <number>
    <field>9961xxxxxx</field>  <!--First field should be mobile no.-->
    <field>Rajeev</field>  <!--FIELD2 Data-->
    <field>12</field>  <!--FIELD3 Data-->
    <field>120</field>  <!--FIELD4 Data-->
  </number>
  <number>
    <field>9020xxxxxx</field>  <!--First field should be mobile no.-->
    <field>Saju</field>  <!--FIELD2 Data-->
    <field>15</field>  <!--FIELD3 Data-->
    <field>150</field>  <!--FIELD4 Data-->
  </number>
</data >
```

Now you need to submit this data into our API, after urlencoding the data, as follows:

```
xml= <data> <uid>your_uid</uid> <pin>your_pin</pin>
<sender>SENDER</sender> <route>0</route> <pushid>>true</pushid>
<dynamic>>true</dynamic> <message> Dear #FIELD2#, Your account has
credited with Rs #FIELD3#. Your account Balance is #FIELD4# </message>
<number> <field>9961xxxxxx</field> <field>Rajeev</field> <field>12</field>
<field>120</field> </number> <number> <field>9400xxxxxx</field>
<field>Saju</field> <field>15</field> <field>150</field> </number> </data>
```

Urlencoded form of the above data is

```
%3Cdata%3E+%3Cuid%3Eyour_uid%3C%2Fuid%3E+%3Cpin%3Eyour_pin%3C%2Fpin%3E+%3Csender%3ESENDER%3C%2Fsender%3E+%3Croute%3E0%3C%2Froute%3E+%3Cpushid%3Etrue%3C%2Fpushid%3E+%3Cdynamic%3Etrue%3C%2Fdynamic%3E+%3Cmessage%3E+Dear+%23FIELD2%23%2C+Your+account+has+credited+with+Rs+%23FIELD3%23.+Your+account+Balance+is+%23FIELD4%23+%3C%2Fmessage%3E+%3Cnumber%3E+%3Cfield%3E9961xxxxxx%3C%2Ffield%3E+%3Cfield%3ERajeev%3C%2Ffield%3E+%3Cfield%3E12%3C%2Ffield%3E+%3Cfield%3E120%3C%2Ffield%3E+%3C%2Fnumber%3E+%3Cnumber%3E+%3Cfield%3E9400xxxxxx%3C%2Ffield%3E+%3Cfield%3ESaju%3C%2Ffield%3E+%3Cfield%3E15%3C%2Ffield%3E+%3Cfield%3E150%3C%2Ffield%3E+%3C%2Fnumber%3E+%3C%2Fdata%3E
```

Now just POST these values to our Server for delivery

```
http://yourdomain.com/api/sms.php?xml=
%3Cdata%3E+%3Cuid%3Eyour_uid%3C%2Fuid%3E+%3Cpin%3Eyour_pin%3C%2Fpin%3E+%3Csender%3ESENDER%3C%2Fsender%3E+%3Croute%3E0%3C%2Froute%3E+%3Cdynamic%3Etrue%3C%2Fdynamic%3E+%3Cmessage%3E+Dear+%23FIELD2%23%2C+Your+account+has+credited+with+Rs+%23FIELD3%23.+Your+account+Balance+is+%23FIELD4%23+%3C%2Fmessage%3E+%3Cnumber%3E+%3Cfield%3E9961xxxxxx%3C%2Ffield%3E+%3Cfield%3ERajeev%3C%2Ffield%3E+%3Cfield%3E12%3C%2Ffield%3E+%3Cfield%3E120%3C%2Ffield%3E+%3C%2Fnumber%3E+%3Cnumber%3E+%3Cfield%3E9400xxxxxx%3C%2Ffield%3E+%3Cfield%3ESaju%3C%2Ffield%3E+%3Cfield%3E15%3C%2Ffield%3E+%3Cfield%3E150%3C%2Ffield%3E+%3C%2Fnumber%3E+%3C%2Fdata%3E
```

System will return you the unique message id as comma separated format which can be saved for future references

232323,2323231

If **<pushid>1</pushid>** is used, then the response will be a single push id which can be saved and used for retrieving the status of all the messages associated with that push id.

1504565

XML API for Bulk Push

Our Simple Yet Powerful Dynamic Messaging API allows bulk pushing different messages to different mobile numbers through a Single API Call.

`http://yourdomain.com/api/sms.php?xml=XML_DATA`

'`xml=XML_DATA`' indicates that you use XML API for bulk push.

Bulk Push XML API can be used for sending different messages to different numbers at a time. This is highly useful in the case of sending different information to people.

Consider an example where you need to send different notifications to the customers of a shop about different products. You can complete this task in a single call to our Bulk Push XML API.

Sample XML Format

```
<data>
  <uid>your_uid</uid>
  <pin>your_pin</pin>
  <sender>SENDER</sender>
  <route>0</route>
  <pushid>1</pushid>
  <bulkpush>1</bulkpush> <!--For enabling bulk push-->
  <number>
    <field>9961xxxxxx</field> <!--Mobile Number-->
    <field>FIRST MESSAGE</field> <!--Actual Message-->
  </number>
  <number>
    <field>9400xxxxxx</field> <!--Mobile Number-->
    <field>SECOND MESSAGE</field> <!--Actual Message-->
  </number>
</data>
```

Now you need to submit this data into our API, after urlencoding the data, as follows:

```
xml= <data> <uid>your_uid</uid> <pin>your_pin</pin>
<sender>SENDER</sender> <route>0</route> <pushid>true</pushid>
<bulkpush>true</bulkpush> <number> <field>9961xxxxxx</field>
<field>FIRST MESSAGE</field> </number> <number>
<field>9400xxxxxx</field> <field>SECOND MESSAGE</field> </number>
</data>
```

Urlencoded form of the above data is

```
%3Cdata%3E+%3Cuid%3Eyour_uid%3C%2Fuid%3E+%3Cpin%3Eyour_pin%3C%2Fpin%3E+%3Csender%3ESENDER%3C%2Fsender%3E+%3Croute%3E0%3C%2Froute%3E+%3Cpushid%3Etrue%3C%2Fpushid%3E+%3Cbulkpush%3Etrue%3C%2Fbulkpush%3E+%3Cnumber%3E+%3Cfield%3E9961xxxxxx%3C%2Ffield%3E+%3Cfield%3EFIRST+MESSAGE%3C%2Ffield%3E+%3C%2Fnumber%3E+%3Cnumber%3E+%3Cfield%3E9400xxxxxx%3C%2Ffield%3E+%3Cfield%3ESEC+MESSAGE%3C%2Ffield%3E+%3C%2Fnumber%3E+%3C%2Fdata%3E
```

Now just POST these values to our Server for delivery

```
http://yourdomain.com/api/sms.php?xml=%3Cdata%3E+%3Cuid%3Eyour_uid%3C%2Fuid%3E+%3Cpin%3Eyour_pin%3C%2Fpin%3E+%3Csender%3ESENDER%3C%2Fsender%3E+%3Croute%3E0%3C%2Froute%3E+%3Cpushid%3Etrue%3C%2Fpushid%3E+%3Cbulkpush%3Etrue%3C%2Fbulkpush%3E+%3Cnumber%3E+%3Cfield%3E9961xxxxxx%3C%2Ffield%3E+%3Cfield%3EFIRST+MESSAGE%3C%2Ffield%3E+%3C%2Fnumber%3E+%3Cnumber%3E+%3Cfield%3E9400xxxxxx%3C%2Ffield%3E+%3Cfield%3ESEC+MESSAGE%3C%2Ffield%3E+%3C%2Fnumber%3E+%3C%2Fdata%3E
```

System will return you the unique message id as comma separated format which can be saved for future references

232323,2323231

If **<pushid>1</pushid>** is used, then the response will be a single push id which can be saved and used for retrieving the status of all the messages associated with that push id.

1504565

API for Scheduling SMS

http://yourdomain.com/api/sms.php?uid=your_uid&pin=your_pin&sender=SENDER&route=0&mobile=MOBILE1&message=MESSAGE&time=dd-mm-yyyy-24h-min

Tips : 4th January 2011 at 2:30 PM (format 04-01-2011-14-30)

You can schedule a message to deliver automatically at any time except (8:45pm - 9:10am). **sender** is optional for scheduling Normal SMS and Priority SMS, but compulsory for Transaction SMS.

Sample Response

On successful scheduling you will receive an unique numeric id like

155246332

If you try to schedule a message in between 8:45pm - 9:10am, response will be You can't schedule messages in between 8:45pm - 9:10am. (Not Applicable for Transactional Route)

API for Deleting Scheduled SMS

http://yourdomain.com/api/deleteschedule.php?uid=your_uid&pin=your_pin&msgid=MESSAGE-ID

If you are passing the pushid then use pushid=Your PUSH-ID

Sample Response

The response will be either

'Message already sent ' if it is already sent

Or

'Scheduled Message Deleted' if the message is correctly deleted.

Fetching Delivery Reports

You can fetch delivery status of each messages based on the unique Message ID or Push ID

Message Id : Whenever you are sending an SMS our system will give a message id for each numbers, which can be saved into your database and in future by calling these message id's you can update the delivery status.

Push Id : All messages are assigned with a unique push id, it is very useful if you are passing more than 1 number through Multiple Sending API or Dynamic Sending API. In a single API call you can get complete delivery status of the numbers associated with the Push.

To get the Push ID as response you have to pass pushid=1 as extra parameter in the SMS Sending API, default response is Message ID

Following are the possible responses you will get when you trigger the Delivery Report API's

Sent = SMS Submitted to SMSC [this status can change]

Delivered = Message is correctly delivered

DND = The receiver number is registered under DND

Failed = SMS Failed and not Delivered

Invalid Number = Invalid Number SMS Not Delivered

Invalid MessageID sent = Incorrect messageID passed

DND Template Mismatch = Message content is not matching with template

API to Fetch Single Delivery Report

http://yourdomain.com/api/dlr.php?uid=your_uid&pin=your_pin&msgid=MESSAGE-ID

Sample Response

Delivered

Fetching Delivery Report based on Push ID

http://yourdomain.com/api/dlr.php?uid=your_uid&pin=your_pin&pushid=PUSH-ID&rtype=json/xml/csv

Sample JSON Response as below

```
[{"smsid": "48", "mobile": "9020xxxxxx", "status": "Sent"}]
```

Sample XML Response as below

```
<reports>
  <report>
    <smsid>48</smsid>
    <mobile>9020xxxxxx</mobile>
    <status>Sent</status>
  </report>
</reports>
```

Sample CSV Response as below

48,9020xxxxxx,Sent

API to Fetch Multiple Delivery Report

http://yourdomain.com/api/dlr.php?uid=your_uid&pin=your_pin&msgid=message_id1,message_id2,message_id3

Sample Response

Suppose you pass the message ids as

234445,254445

API response, each separated by comma(delimiter), will be

Sent,Sent	-Both messages are submitted to SMSC
Sent,Delivered	-First one sent, second Message is correctly delivered
DND,Sent	-The first receiver number is registered under DND, second message sent

API to Fetch Multiple Delivery Report using PushId

http://yourdomain.com/api/dlr.php?uid=your_uid&pin=your_pin&pushid=PUSH-ID&rtype=json/xml/csv

Sample JSON Response as below

```
[{"smsid": "49", "mobile": "9020xxxxxx", "status": "Sent"}, {"smsid": "50", "mobile": "9037xxxxxx", "status": "Sent"}]
```

Sample XML Response as below

```
<reports >
  <report >
    <smsid >48</smsid >
    <mobile >9020xxxxxx</mobile >
    <status >Sent </status >
  <report >
  <report >
    <smsid >48</smsid >
    <mobile >9037xxxxxx</mobile >
    <status >Sent </status >
  <report >
</reports >
```

Sample CSV Response as below

```
49,9020xxxxxx,Sent <new line >
50,9037xxxxxx,Sent <new line >
```

API to Fetch Datewise Delivery Report

`http://yourdomain.com/api/dlr.php?uid=your_uid&pin=your_pin&date=dd-mm-yyyy&route=0/1/5&rtype=json/xml/csv`

route=0 -> Normal Route
route=1 -> Priority Route
route=4 -> Marketing Route
route=5 -> Transaction Route

Optional Parameters

`rtype=json/xml/csv`

`route=0/1/4/5`

Sample JSON Response as below

```
[{"smsid": "48", "mobile": "9020xxxxxx", "status": "Sent"}, {"smsid": "49", "mobile": "8547xxxxxx", "status": "Sent"}, {"smsid": "50", "mobile": "9037xxxxxx", "status": "Sent"}]
```

Sample XML Response as below

```
<reports >
  <report >
    <smsid >48</smsid >
    <mobile >9020xxxxxx</mobile >
    <status >Sent</status >
  <report >
  <report >
    <smsid >48</smsid >
    <mobile >9020xxxxxx</mobile >
    <status >Sent</status >
  <report >
</reports >
```

Sample CSV Response as below

```
48,9020xxxxxx,Sent <new line >
49,8547xxxxxx,Sent <new line >
50,9037xxxxxx,Sent <new line >
```

API to Fetch Current Normal Balance

http://yourdomain.com/api/balance.php?uid=your_uid&pin=your_pin&route=0

Sample Response

The response for balance checking will be the total count of normal messages remaining in your account.

10224

API to Fetch Current Priority Balance

http://yourdomain.com/api/balance.php?uid=your_uid&pin=your_pin&route=1

Sample Response

The response for balance checking will be the total count of priority messages remaining in your account.

1022423

API to Fetch Current Marketing Balance

http://yourdomain.com/api/balance.php?uid=your_uid&pin=your_pin&route=4

Sample Response

The response for balance checking will be the total count of marketing messages remaining in your account.

1022434

API to Fetch Current Transaction Balance

http://yourdomain.com/api/balance.php?uid=your_uid&pin=your_pin&route=5

Sample Response

The response for balance checking will be the total count of transaction messages remaining in your account.

1022434

API to List All Sender ID's

http://yourdomain.com/api/getsenderids.php?uid=your_uid&pin=your_pin

All sender id's will be returned in comma separated format .

Sender Ids are allocated under special procedures only. They are 6 DIGIT alpha codes issued after proper verification.

Sample Response

Example : Sender1,Sender2,Sender3,Sender4,

All Possible Responses & Errors

Unique numeric id	API normal Response after processing
0,No balance	No SMS left in your account
Wrong UserID or PIN	You have entered either UserID or PIN incorrectly
0,Invalid SenderID	(For transactional SMS only) Sender ID is wrong
0,DND Template Not Matching	DND_Template ID incorrect or mismatch
You can't schedule messages in between 8:45pm - 9:10am	It is not possible to schedule a message in between 8:45pm- 9.10am
Message already sent	You cannot delete the scheduled message as it is already sent from our system.
Scheduled Message Deleted	The sheduled message is deleted.
Invalid Message IDSent	The Message ID is wrong, so you can't track the message with this ID
Sent	The message submitted to SMSC for delivery, not yet acknowledged by the end service provider.
Delivered	The message is delivered to the intended user.
DND	You cannot send message to this number since it is registered under NDNC.

Failed	SMS Failed, Not Delivered
DND Template Mismatch	DND Template not matched, SMS Not Delivered.
Sender1,Sender2,Sender3,Sender4,	Lists all sender Ids available with your account
Invalid Request	Missing a compulsory parameter

Description of Files Used

sms.php	This file handles all the 'send sms' requests. It's function includes handling send requests through all the different routes ,multiple sending or dynamic sending and scheduling.
balance.php	This file processes the balance enquiry requests for all the routes. You just need to pass your UID and PIN into this file.
dlr.php	Whenever you want to access the delivery reports of the messages you sent, you need this file. This file helps in fetching single or multiple delivery reports and also datewise delivery report. You have to pass either message ID or push ID to get this done. To get datewise delivery report, pass the date in a specified format along with other optional parameters.
deleteschedule.php	This file is exclusively used for deleting the scheduled messages in your account.
getsenderids.php	For Transaction SMS , you need to pass your sender ID. To list all the sender Ids associated with your account, you only need to pass your UID and PIN.
template.php	You can request any number of Transactional Message contents to be used in your transactional gateway. You need to pass the message content to this file after 'urlencoding'.

Sample API Calling Scripts

Sample API calling source script of PHP, ASP, ASP.NET, C# is available from our website from the API & Downloads -> Downloads Section from

We have PHP Library for Sending SMS through Simple Functions .Please download the same from the API & Downloads -> Downloads

Important Instructions

Don't copy paste any contents from any HTML or Web Pages, if you need to copy paste then first copy the contents to Notepad then Paste into our textbox

Before submitting messages to our server please make sure that you have encoded the content of the message using urlencode function

Before start integrating our API into your application please construct an API and call the API in any browser and make sure that you got correct response

Example : If you would like to send an SMS using our priority gateway then just fill all the required parameter of the api and copy paste into your browser address bar and hit Enter.

URL Encoding Details

We always suggest you to encode the content of the message before submitting into our server which ensures that messages is correctly displayed in the mobile handset

if you are not aware about "urlencode" please contact your programmer or please go through the below websites to have an exact idea.

http://www.w3schools.com/tags/ref_urlencode.asp

Online URL Encoder & Decoder

<http://meyerweb.com/eric/tools/dencoder/>

Example before URL Encode

Hello this is a test text before url encode

Example after URL Encode

Hello %20this %20is %20a %20test %20text %20before %20url %20encode %20

Sample API Call (Transaction Route)

http://yourdomain/api/sms.php?uid=your_uid&pin=your_in&sender=SENDERID &route=5&tempid=DND_TEMPLATE_ID&mobile=MOBILE1&message=Hello %20thi s %20is %20a %20test %20text %20before %20url %20encode %20