

HTTP/XML/JSON API V4

USER MANUAL

Contents

[Messaging API 9](#_Toc487645948)

[How does this work? 9](#_Toc487645949)

[Base URL for API Integration 9](#_Toc487645950)

[Authentication 10](#_Toc487645951)

[Required Environment 10](#_Toc487645952)

[SSL Enabled API Triggering 11](#_Toc487645953)

[Output Restrictions 11](#_Toc487645954)

[Input Restrictions 11](#_Toc487645955)

[Pre requisites for API Integration 12](#_Toc487645956)

[Mobile Phone Number Format 12](#_Toc487645957)

[Sender ID 12](#_Toc487645958)

[Creating a Sender ID (Alphabetic Originator) 12](#_Toc487645959)

[Supported Character Sets 12](#_Toc487645960)

[GSM 03.38 Encoding 12](#_Toc487645961)

[Unicode 13](#_Toc487645962)

[Message Length - Multipart Messages 13](#_Toc487645963)

[Standard GSM 03.38 Character Set 13](#_Toc487645964)

[Unicode Characters 14](#_Toc487645965)

[Delivery URL 14](#_Toc487645966)

[Custom Flags in DLRURL 15](#_Toc487645967)

[Sample DLRURL 15](#_Toc487645968)

[Without Encode 16](#_Toc487645969)

[API Responses 16](#_Toc487645970)

[Successful Submission of SMS 16](#_Toc487645971)

[Response Formats 16](#_Toc487645972)

[Response Parameters 16](#_Toc487645973)

[API Error Codes 17](#_Toc487645974)

[SMS Status Codes- HTTP API (On Submission) 19](#_Toc487645975)

[SMS Status Codes-XML/JSON API (On Submission) 20](#_Toc487645976)

[SMS Status Codes (Operator Level) 20](#_Toc487645977)

[Sending an SMS 22](#_Toc487645978)

[API Parameter Specifications 22](#_Toc487645979)

[Mandatory Parameters 23](#_Toc487645980)

[Optional Parameters 23](#_Toc487645981)

[Advanced Parameters(Optional) 24](#_Toc487645982)

[Example Responses 24](#_Toc487645983)

[JSON 24](#_Toc487645984)

[XML 25](#_Toc487645985)

[Error Responses 26](#_Toc487645986)

[Response Data 26](#_Toc487645987)

[Send a Unicode SMS 27](#_Toc487645988)

[Mandatory Parameters 27](#_Toc487645989)

[Schedule an SMS 27](#_Toc487645990)

[API Parameter Specifications 27](#_Toc487645991)

[Mandatory Parameters 28](#_Toc487645992)

[Sending a Port SMS 28](#_Toc487645993)

[Mandatory Parameters 29](#_Toc487645994)

[Send an SMS using XML API 29](#_Toc487645995)

[Sample XML Data 29](#_Toc487645996)

[API Parameter Specifications 30](#_Toc487645997)

[Mandatory Parameters 30](#_Toc487645998)

[Optional Parameter 31](#_Toc487645999)

[XML Tag Parameters 31](#_Toc487646000)

[Example Responses 34](#_Toc487646001)

[JSON 34](#_Toc487646002)

[Send an SMS Using JSON API 35](#_Toc487646003)

[Sample JSON data 35](#_Toc487646004)

[API Parameter Specifications 36](#_Toc487646005)

[Mandatory Parameters 36](#_Toc487646006)

[Optional Parameter 36](#_Toc487646007)

[JSON Tag Parameters 37](#_Toc487646008)

[Sample Response 40](#_Toc487646009)

[JSON 40](#_Toc487646010)

[Checking SMS Status -PULL 41](#_Toc487646011)

[API Parameter Specifications 41](#_Toc487646012)

[Mandatory Parameters 41](#_Toc487646013)

[Optional Parameters 41](#_Toc487646014)

[Example Responses 42](#_Toc487646015)

[JSON 42](#_Toc487646016)

[Error Response 42](#_Toc487646017)

[Response Data 43](#_Toc487646018)

[Sending SMS to an Opt In Group 44](#_Toc487646019)

[API Parameter Specifications 45](#_Toc487646020)

[Mandatory Parameters 45](#_Toc487646021)

[Optional Parameters 46](#_Toc487646022)

[Example Responses 46](#_Toc487646023)

[JSON 46](#_Toc487646024)

[Expected Error Codes: 48](#_Toc487646025)

[Error Response 48](#_Toc487646026)

[Response Data 49](#_Toc487646027)

[Create a Group 50](#_Toc487646028)

[API Parameter Specifications 50](#_Toc487646029)

[Mandatory Parameters 50](#_Toc487646030)

[Optional Parameter 50](#_Toc487646031)

[Example Responses 51](#_Toc487646032)

[JSON 51](#_Toc487646033)

[Error Response 51](#_Toc487646034)

[Response Data 51](#_Toc487646035)

[Add Contacts to a group 51](#_Toc487646036)

[API Parameter Specifications 52](#_Toc487646037)

[Mandatory Parameters 52](#_Toc487646038)

[Optional Parameters 52](#_Toc487646039)

[Example Responses 53](#_Toc487646040)

[JSON 53](#_Toc487646041)

[Error Response: 53](#_Toc487646042)

[API Error Codes 53](#_Toc487646043)

[Response Data 54](#_Toc487646044)

[Send Message to a group 54](#_Toc487646045)

[API Parameter Specifications 54](#_Toc487646046)

[Mandatory Parameters 55](#_Toc487646047)

[Example Responses 55](#_Toc487646048)

[JSON 55](#_Toc487646049)

[Error Response 56](#_Toc487646050)

[Response Data 56](#_Toc487646051)

[API Error Codes 57](#_Toc487646052)

[Delete Scheduled SMS 57](#_Toc487646053)

[API Parameter Specifications 58](#_Toc487646054)

[Mandatory Parameters 58](#_Toc487646055)

[Example Responses 58](#_Toc487646056)

[JSON 58](#_Toc487646057)

[Error Response 58](#_Toc487646058)

[Response Data 59](#_Toc487646059)

[Edit Schedule 59](#_Toc487646060)

[API Parameter Specifications 59](#_Toc487646061)

[Mandatory Parameters 59](#_Toc487646062)

[Optional Parameters 60](#_Toc487646063)

[Example Responses 60](#_Toc487646064)

[JSON 60](#_Toc487646065)

[Response Data 61](#_Toc487646066)

[Expected API Error Codes: 61](#_Toc487646067)

[Check Credits Availability 61](#_Toc487646068)

[API Parameter Specifications 61](#_Toc487646069)

[Mandatory Parameters 62](#_Toc487646070)

[Optional Parameters 62](#_Toc487646071)

[Example Responses 62](#_Toc487646072)

[JSON 62](#_Toc487646073)

[Error Response 62](#_Toc487646074)

[Response Data 63](#_Toc487646075)

[HLR Request 63](#_Toc487646076)

[API Parameter Specifications 63](#_Toc487646077)

[Mandatory Parameters 64](#_Toc487646078)

[Example Responses 64](#_Toc487646079)

[JSON 64](#_Toc487646080)

[Error Response 65](#_Toc487646081)

[Response Data: 65](#_Toc487646082)

[Lookup using XML/JSON Data 65](#_Toc487646083)

[API Parameter Specifications 65](#_Toc487646084)

[Mandatory Parameters 66](#_Toc487646085)

[Optional Parameter 67](#_Toc487646086)

[Sample JSON Data 67](#_Toc487646087)

[Sample XML Data 67](#_Toc487646088)

[Custom XML/JSON Tags 68](#_Toc487646089)

[Example Responses 69](#_Toc487646090)

[JSON 69](#_Toc487646091)

[XML 70](#_Toc487646092)

[Response Data: 71](#_Toc487646093)

[Sample DLRURL 71](#_Toc487646094)

[Creating a Txtly 71](#_Toc487646095)

[API Parameter Specifications 72](#_Toc487646096)

[Mandatory Parameters 72](#_Toc487646097)

[Optional Parameters 72](#_Toc487646098)

[Example Responses 73](#_Toc487646099)

[JSON 73](#_Toc487646100)

[Error Response 73](#_Toc487646101)

[Response Data 73](#_Toc487646102)

[Expected Error Codes 74](#_Toc487646103)

[Deleting a Txtly 74](#_Toc487646104)

[API Parameter Specifications 74](#_Toc487646105)

[Mandatory Parameters 75](#_Toc487646106)

[Optional Parameters 75](#_Toc487646107)

[Example Responses 76](#_Toc487646108)

[JSON 76](#_Toc487646109)

[Error Response 76](#_Toc487646110)

[Response Data 76](#_Toc487646111)

[Txtly Reports Extraction - PULL 77](#_Toc487646112)

[API Parameter Specifications 77](#_Toc487646113)

[Mandatory Parameters 77](#_Toc487646114)

[Optional Parameter 78](#_Toc487646115)

[Example Responses 78](#_Toc487646116)

[JSON 78](#_Toc487646117)

[Pull logs for Individual Txtly 81](#_Toc487646118)

[API Parameter Specifications 81](#_Toc487646119)

[Mandatory Parameters 81](#_Toc487646120)

[Example Responses 82](#_Toc487646121)

[JSON 82](#_Toc487646122)

[Send an Email 84](#_Toc487646123)

[Send Simple Text Email 84](#_Toc487646124)

[API Parameter Specifications 84](#_Toc487646125)

[Mandatory Parameters 84](#_Toc487646126)

[Optional Parameters 85](#_Toc487646127)

[Example Responses 86](#_Toc487646128)

[JSON 86](#_Toc487646129)

[XML 87](#_Toc487646130)

[Send An Email With Attachment 88](#_Toc487646131)

[Mandatory Parameters 88](#_Toc487646132)

[Optional Parameters 89](#_Toc487646133)

[Send an Email using Template 90](#_Toc487646134)

[Mandatory Parameters 90](#_Toc487646135)

[Optional Parameters 91](#_Toc487646136)

[Send an HTML email 92](#_Toc487646137)

[Mandatory Parameters 93](#_Toc487646138)

[Optional Parameters 94](#_Toc487646139)

[Expected API Error Codes 96](#_Toc487646140)

[Developers Guide 97](#_Toc487646141)

[PHP 97](#_Toc487646142)

[JAVA 98](#_Toc487646143)

[ASP.NET 99](#_Toc487646144)

## Base URL for API Integration

Base URL will be defined as the common API URL parameters to be specified in every API. All APIs shall contain this base URL.

|  |
| --- |
| https://domain\_URL/api/v4/?api\_key=Ad9e5XXXXXXXXXXXXX |

The following are the API Parameters

|  |  |  |
| --- | --- | --- |
| Parameter | Description | Expected Values |
| api\_key | API/Working key generated from your SMS Account | Alphanumeric working key generated from web panel |

## Authentication

All requests are secured through BASIC authentication through API Key (api\_key) which will be provided by Messaging service provider only.

An additional level of protection is provided where API key can be configured to accept requests from a specific IP Address only. Also, API usage with username & password combination is removed due to security concerns.

One way to secure a web-based application is to restrict access based on the IP address. You can block access to a specific address or range of addresses that you suspect belong to malicious individuals.

## Required Environment

To use API, one need to have a system that has internet connectivity. This may be direct internet connectivity or via a firewall or proxy server. Additionally, if anyone wish to receive delivery reports (DLRs), then one must have a web server that is able to receive incoming requests from the Internet. This web server does not necessarily have to be running on the same machine that is triggering messages to us.

## SSL Enabled API Triggering

These APIs has been designed to allow client to access them using an SSL Enabled connection for added security. To trigger API on an SSL enabled connection, one must enter the URL beginning with https:// instead of http://

Our API supports both HTTP and HTTPS. Using HTTPS to encrypt all requests through SSL is recommended, but if you're using a client and tool that's not setup to make SSL requests, one can use HTTP also.

SSL encryption ensures that all data passed between web server and browser remains private and integral.

## Output Restrictions

If the provider network is down, our gateway Servers will retry your message up to six hours before depletion.

1. Duplicate messages in single request only are filtered by default.
2. A "duplicate message" is defined as the same destination address, sender address and the message body.
3. Duplicate messages sent by different request are NOT filtered.

## Input Restrictions

This is one of the simpler server-based method of communication. This can be used either in the form of a HTTP POST or HTTP GET. We recommend POST for larger data transfer and data security.

1. All calls to the API must be URL-encoded.
2. The parameter variables are case sensitive.
3. We recommended to use the POST method for error free submission due to URL limitations.

# Pre requisites for API Integration

## Mobile Phone Number Format

All mobile phone numbers should be a valid with country code for global messaging with or without country code for domestic platform. For eg: +919xxxxxxxxx, 918xxxxxxxxx, 7xxxxxxxxx.

## Sender ID

All text messages carry a sender ID. When you send a text message through API, you could use any of the approved sender IDs for your account. You can request for a new sender ID right from your SMS application.

### Creating a Sender ID (Alphabetic Originator)

One can create the sender ID using company name, or anything else. This sender ID will be shown as SMS sender name to the recipients rather than just a phone number in the sender field.

* The alphabetic originator, or sender ID, can contain characters from a-z, A-Z. Other characters may cause message delivery to be failed. Every country has their own restricted variations for defining sender IDs.

## Supported Character Sets

### GSM 03.38 Encoding

The standard SMS mobile-phone character set is used universally to send text messages. The standard character set allows few of the Roman alphabet and the most common accented forms, certain special characters, and few Greek alphabets that sum up to 160 characters. It includes the following characters:

|  |
| --- |
| @ £ \ $ ¥ è é ù ì ò Ç \ n Ø ø \ r Å å ? \_ F " ? O ? ? S T ? Æ æ ß É ! \ " # ¤ % & ' ( ) \* + , - ../ 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 ? Ö Ñ Ü § ¿ 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 ö ñ ü à ä Ä { } \ ~ [ ] | |

* Note: All extended characters will take two characters to send.
* Extended characters include:

|  |
| --- |
| £ ¥ è é ù ì ò Ç Ø ø Å å Δ Φ Γ Λ Ω Π Ψ Σ Θ Ξ Æ æ ß É ¤ ¡ Ä Ö Ñ Ü § ¿ ä ö ñ ü à |

* Also, extended character like € takes three characters to send.

### Unicode

While using non-GSM, 03.38 characters SMS transmits as a Unicode message. The Unicode character support provides the ability to send a message in Hindi, Kannada, Tamil, etc.

If the message to be transmitted is in Unicode format, then entire message has to be converted into Unicode format. Also, Unicode transmission will charge double the credits, including normal English SMS sent in Unicode mode.

## Message Length - Multipart Messages

### Standard GSM 03.38 Character Set

Single text messages are limited to 160 characters, including spaces. The maximum length of two-part or three-part multipart text messages is 306 (2 x 153) and 459 (3 x 153) characters, respectively. One can send up to 1000 characters as multipart message which will be split into 7 messages internally. This refers only to messages that contain the standard GSM 03.38-character set. Each multipart text message is limited to 153 characters rather than 160 due to the need for user-data headers (UDHs) information. Mobile phones use UDH information to enable them to link long messages together so that they appear as single SMS messages in recipient’s phone inbox.

|  |  |
| --- | --- |
| Number of SMS | Number of characters in the linked SMS |
| 1 | 160 characters |
| 2 | 306 characters (2 x 153) |
| 3 | 459 characters (3 x 153) |
| ... | .. |
| 7 | upto 1000 characters (6 x 153) + (1 x 82) |

### Unicode Characters

Using Unicode, for languages such as Hindi, restricts your message to a maximum of 70 characters per SMS. You can, however, send longer text messages, with our system splitting them into two or at the three separate text messages. The maximum lengths of two-part and three-part multipart Unicode text messages are 134 (2 x 67) and 201 (3 x 67) characters, respectively. You could send up to 500 characters as multipart message which will be split into 8 messages internally.

|  |  |
| --- | --- |
| Number of SMS | Number of characters in the linked SMS |
| 1 | 70 characters |
| 2 | 134 characters (2 x 67) |
| 3 | 201 characters (3 x 67) |
| 8 | 500 characters (7 x 67) + (1 x 31) |
| ... | .. |

## Delivery URL

DLRURL is basically the URL where you can get the delivery reports if requested via API. To request such delivery reports, you need to pass "DLRURL" parameter in the API URL.

## Custom Flags in DLRURL

We have few custom flags supported in DLR URL to get the additional values. In case one wants to build the URL in above format, the following flags can be used.

|  |  |
| --- | --- |
| Field Name | Description |
| {status} | SMS status code |
| {sid} | MessageID generated by us |
| {custom} | ReferenceID provided for the sms |
| {custom1} | Custom Flag 1 associated with the message |
| {custom2} | Custom Flag 2 associated with the message |
| {credits} | Credits charged for the Sms |
| {senttime} | Sent to Operator time in YYYY-MM-DD HH:MM:SS format |
| {submittime} | Request receive time in YYYY-MM-DD HH:MM:SS format |
| {delivered} | Delivered time in YYYY-MM-DD HH:MM:SS format |
| {sentat} | Sent to Operator in unixtime format |
| {delivat} | Delivered time in unixtime format |
| {submitat} | Request Receive time in unixtime format |
| {mobile} | Mobile Number |
| {units} | No of Message Units |

### Sample DLRURL

|  |
| --- |
| http%3A%2F%2Fdomain.net%2Freceive%3Fsent%3D%7Bsent%7D%26delivered%3D%7Bdelivered%7D%26custom%3D%7Bcustom%7D%26sid%3D%7Bsid%7D%26status%3D%7Bstatus%7D%26reference%3D%7Breference%7D%26custom1%3D%7Bcustom1%7D%26custom2%3D%7Bcustom2%7D%26credits%3D%7Bcredits%7D%0A |

### Without Encode

|  |
| --- |
| http://domain.net/receive?sent={sent}&delivered={delivered}&custom={custom}&sid={sid}&status={status}&reference={reference}&custom1={custom1}&custom2={custom2}&credits={credits} |

## API Responses

## Successful Submission of SMS

For each successful submission of an SMS, the API would return a unique message ID (UUID) for that SMS campaign. The Client’s Application should capture the unique message ID and fetch a Delivery Report (DLR) against that corresponding message ID. This message ID then can be used to track the status of the message and receive the delivery report against this message ID through the delivery report URL specified by the client.

## Response Formats

By default, the API URL generate responses in JSON format. One can also generate responses in XML, JSON formats. All we need to do is to add "&format=xml" (for XML format).

## Response Parameters

The response parameters in case of error in API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| status | Error code for your API call | Numeric Value A401,E601 etc |
| message | Error Message for your API call | String.Refer API Error codes section |

## API Error Codes

When certain parameters of the API URL are not entered correctly, you will receive a response from the URL stating the type of error. These error descriptions will help you to pinpoint the erroneous parameter in the URL and rectify it.

Following is the aggregated list of error codes which can be generally received while using the API in the below format:

|  |  |
| --- | --- |
| **Status** | **Message** |
| A400 | Api file not found in project |
| A401A | Method Not Implemented. |
| A401B | Method Not Found |
| A402 | Api Key not found. |
| A403 | Api Signature not found |
| A404 | Your api account got suspended |
| A405 | Api secret not found. |
| A406 | Request is not allowed from this ip |
| A407 | Header misconfigured |
| A408 | Api Signature not equal |
| A411 | Port number must be numeric |
| A415 | Invalid xml/Invalid Json |
| A416 | Invalid Message ID/Mobile |
| A417 | Invalid Message ID/Mobile |
| A418 | Invalid Group ID/Campaign |
| A419 | Invalid Group ID/Campaign |
| A430 | Invalid Group ID |
| A431 | Invalid Group ID |
| A432 | Campaign already completed/Campaign already cancelled |
| A433 | Schedule time already passed |
| A434 | Campaign within next 5 minutes cannot be modified/cancelled. |
| A434 | Invalid schedule time. Minimum 5 min time gap is required and 3 months from now. |
| E601 | Missing recipient |
| E603 | Missing message |
| E602 | Missing sender ID |
| E604 | SenderID must be less than 14 chars |
| E605 | Invalid Sender or Sender not allowed |
| E606 | Invalid schedule time. Minimum 5 min time gap is required and 3 months from now. |
| E606A | Batch size should be greater or equal to 1000 |
| E606B | Schedule minute should be in multiples of 15. Minimum 15 minutes |
| E612 | Invalid Template Matched |
| E607 | SMS length is exceeding 1000 characters |
| E607A | Unable to find gateway to sent sms. |
| E608 | No Valid mobile numbers found |
| E608A | Your daily usage limit exceeded |
| E609 | An error occurred while saving group details |
| E610 | Unable to write to temp files |
| E611 | Unable to process your request. try again |
| A420 | Group name/id missing |
| A421 | Group is empty/Invalid group |
| A422 | Invalid mobile number |
| A423 | Group name/number missing |
| A424 | Group Name does not exist! |
| A425 | Number does not Exist |
| H601 | Insufficient credits. Total "x". Required "y" more credits |
| A430 | Group name/id missing |
| A431 | Group is empty |
| A432 | Invalid mobile number |
| A433 | Keyword name/number missing |
| A434 | Keyword not found |
| A435 | Keyword is not active |
| A436 | No sender found! |
| E500 | Please provide URL to redirect |
| E501 | Txtly Already Exists. Please try another one. |
| E502 | Please attach a file |
| E503 | Invalid file format. Upload valid file. |
| E504 | File size exceed. Maximum 50000 Kb. |
| E505 | File upload failed. Please try again. |

## SMS Status Codes- HTTP API (On Submission)

When we receive an API call for sending SMS one of the following status code is attached at the application level.

The common status codes received while using the API are:

|  |  |
| --- | --- |
| **Status** | **Description** |
| AWAITED-DLR | Mobile number has been accepted and submitted to the operator. Awaiting delivery report from the operator |
| INVALID-NUM | In case any invalid number present along with the valid numbers. |

## SMS Status Codes-XML/JSON API (On Submission)

There could be one or more nodes in an XML API, each of those nodes can have a different status code on submission. The common status codes received while using the API are:

|  |  |
| --- | --- |
| **Status** | **Description** |
| AWAITED-DLR | Mobile number has been accepted and submitted to the operator.Awaiting delivery report from the operator |
| DNDNUMB | Mobile number is DND. |
| OPTOUT-REJ | Opt out from subscription. |
| INVALID-NUM | In case any invalid number present along with the valid numbers. |
| SENDER-ID-NOT-FOUND | In case the sender ID given in any one of the nodes is not available |
| INV-TEMPLATE- MATCH | In case the message given in any one of the nodes does not match the template approved for the sender ID in the same node |
| MAX-LENGTH | In case the message given in any one of the nodes exceeds the maximum of 1000 characters |

### SMS Status Codes (Operator Level)

On submission of a message to any operator, we associate one of the following status codes to every SMS to track the status. The common status codes received while using any API are as follows:

|  |  |
| --- | --- |
| Status | Description |
| NO-CREDITS | Insufficient credits. |
| SERIES-BLOCK | Mobile number series blocked. |
| INV-NUMBER | Invalid number. |
| SERVER-ERR | Server error |
| SPAM | Spam SMS |
| SNDRID-NOT-ALLOTED | Sender ID not allocated. |
| BLACKLIST | Blacklisted number |
| TEMPLATE-NOT-FOUND | Template not mapped. |
| INV-TEMPLATE-MATCH | Template not matching approved text. |
| SENDER-ID-NOT-FOUND | Sender ID not found. |
| DNDNUMB | DND registered number. |
| NOT-OPTIN | Not subscribed for opt-in group. |
| TIME-OUT-PROM | Time out for promotional SMS. |
| DELIVRD | SMS successfully delivered. |
| INVALID-SUB | Number does not exist. / Failed to locate the number in HLR database. |
| ABSENT-SUB | Telecom services not providing service for the particular number. / Mobile Subscriber not reachable. |
| HANDSET-ERR | Problem with Handset or handset failed to get the complete message. / Handset doesn’t support the incoming messages. |
| BARRED | End user has enabled message barring system. / Subscriber only accepts messages from Closed User Group [CUG]. |
| NET-ERR | Subscriber’s operator not supported. / Gateway mobile switching error. |
| MEMEXEC | Handset memory full. |
| FAILED | SMS expired due to roaming limitation. / Failed to process the message at operator level. |
| MOB-OFF | Mobile handset in switched off mode. |
| HANDSET-BUSY | Subscriber is in busy condition. |
| SERIES-BLK | Series blocked by the operator. |
| EXPIRED | SMS expired after multiple re-try. |
| REJECTED | SMS Rejected as the number is blacklisted by operator. |
| OUTPUT-REJ | Unsubscribed from the group. |
| REJECTED-MULTIPART | Validation fail [SMS over 160 characters] |
| UNDELIV | Failed due to network errors. |
| NO-DLR-OPTR | In case operator has not acknowledged on status report of the SMS. |

# Sending an SMS

HTTP API sends a text message using POST and form-encoded data. The URL you use to send a text message has a required field called api\_key, which is unique and assigned to your account.

The other required parameters for the Request are method, sender, to and message, which are used to define the destination phone number and the message body. If you want to send to multiple phone numbers, provide the numbers in comma separated format.

To send an SMS using the HTTP API, the following URL format is used:

|  |
| --- |
| curl-X POST \ '<BASE\_URL>&method=sms&message=Test+sms&to=9172xxxxxxxx&sender=BRAND |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | BASE URL of your SMS Service | URL |
| method | Predefined method | sms |
| sender | Sender ID assigned to your account | Sender ID |
| to | Receiver Mobile number to which SMS needs to be sent. It can be with or without 91. Also provide multiple numbers in comma separated format. | Mobile number (GET-10 numbers, POST-100 numbers) |
| message | Message to be sent | Message text which is URL encoded (1000 char for normal, 500 for Unicode) |

## Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| dlrurl | The encoded URL to receive delivery reports. Spiffing custom in the DLR url is mandatory. | delivery reports URL |
| custom | Custom message ID by customer, comma separated | Alphanumeric (max 40 char) |
| time | Date and time for scheduling an SMS | EX Format: YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM |
| unicode | To specify that the message to be sent is in unicode format. Also can be used for automatic detection of unicode SMS. | 1 / 0 / auto |
| flash | To specify that the message is to be sent in the flash format | 1 or 0 |

## Advanced Parameters(Optional)

The advanced parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/JSON/JSONP. Default response will be in JSON |
| callback | Callback function for JSONP response format | String |
| custom1 and custom2 | Custom reference fields | Alphanumeric (max 40 char) |
| port | Port number to which SMS has to be sent | Valid integer port number above 2000 |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "data": [  {  "id": "811436f5-0053-4ea6-8cac-b035880b8473:1",  "customid": "1",  "customid1": "11",  "customid2": "22",  "mobile": "95XXXXXXXXX",  "status": "AWAITED-DLR"  },  {  "id": "811436f5-0053-4ea6-8cac-b035880b8473:2",  "customid": "2",  "customid1": "1",  "customid2": "2",  "mobile": "97XXXXXXXX",  "status": "AWAITED-DLR"  }  ],  "message": "Campaign of 2 numbers Submitted successfully." } |

### XML

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" ?> <api>  <status>OK</status>  <data>  <sms>  <id>e6bf8207-d226-4429-a1f0-ba0506efa539:1</id>  <customid>1</customid>  <customid1>11</customid1>  <customid2>22</customid2>  <mobile>95XXXXXXXX</mobile>  <status>AWAITED-DLR</status>  </sms>  <sms>  <id>e6bf8207-d226-4429-a1f0-ba0506efa539:2</id>  <customid>2</customid>  <customid1>1</customid1>  <customid2>2</customid2>  <mobile>99XXXXXXXX</mobile>  <status>AWAITED-DLR</status>  </sms>  </data>  <message>Campaign of 2 numbers Submitted successfully.</message> </api> |

## Error Responses

|  |
| --- |
| {  "status":"A404",  "message": "Your api account got suspended" } |

## Response Data

|  |  |
| --- | --- |
| Parameter | Description |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |
| group\_id | Website group id for your reference. Which is the current date |
| data | All the response is wrapped in side data |
| sms | Each number will be wrapped inside sms |
| id | Website message id for your reference |
| customid | Custom message id if you provide one |
| customid1 | Custom message id 1 if you provide one |
| customid2 | Custom message id 2 if you provide one |
| mobile | Receiver Mobile number |
| status | status of the message |

## Send a Unicode SMS

Unicode SMS is basically a message which can be in regional language. In India, SMS can be sent in Tamil, Kannada, Hindi, English, etc type of languages. To send a unicode SMS using the HTTP API, the following URL format is used:

API Parameter Specifications

Additional API parameter specifications are described in the following sections.

|  |
| --- |
| curl -X POST \'<BASE\_URL>&method=sms&message=Test+sms&to=9172xxxxxxxx&sender=BRAND&unicode=1' |

## Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| Parameter | Description | Expected |
| unicode | To specify that the message to be sent is in unicode format. Also, can be used for automatic detection of unicode SMS. | 1 / 0 / auto |

## Schedule an SMS

This API can let user schedule SMS to single as well as multiple recipients at anytime. SMS cannot be scheduled before 5 minutes and after 3 months from the date of scheduling the SMS. To schedule an SMS using the HTTP API, the following URL format is used:

## API Parameter Specifications

Additional API parameter specifications are described in the following sections.

|  |
| --- |
| curl -X POST \ '<BASE\_URL>&method=sms&message=Test+sms&to=9172xxxxxxxx&sender=BRAND&time=dd%2Fmm%2Fyyyy%20hh%3Amm%20PM' |

## Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| time | Date and time for scheduling an SMS | EX Format: YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM |

## Sending a Port SMS

Using the below API, one can send a normal or unicode SMS to a particular port. Application port numbers can be used in SMS messages to deliver a particular SMS to an application on a mobile phone. If any application is present and running in that particular port, the text will be received as port SMS else the message will hit the native inbox of the mobile phone.

NOTE: The API tries to hit the application(port) for 10 seconds, if unsuccessful the message is delivered on the native inbox

To send a port SMS using the HTTP API, the following URL format is used:

API Parameter Specifications

The API parameter specifications are described in the following sections.

|  |
| --- |
| curl -X POST \ '<BASE\_URL>&method=sms&message=Test+sms&to=9172xxxxxxxx&sender=BRAND&port=8xxx' |

## Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Descriptions** | **Values** |
| port | Port number to which SMS has to be sent | Valid integer port number above 2000 |

# Send an SMS using XML API

To send an SMS using XML API, the following format is used:

|  |
| --- |
| 'curl -X POST \ '<BASE\_URL>&method=sms.xml&xml=%3C%3Fxml%20version%3D%221.0%22%20encoding.....' |

Below is the sample xml data before encoding. You have to send below xml in post method.

## Sample XML Data

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?> <xmlapi>  <sender>AAAAAA</sender>  <message>xml test</message>  <unicode>1</unicode>  <flash>1</flash>  <campaign>xml test</campaign>  <dlrurl>  <![CDATA[http://domain.com/receive?sent={sent}&delivered={delivered}&msgid={msgid}&sid={sid}&status={status}&reference={reference}  &custom1={custom1}&custom2={custom2}&credits={credits}]]>  </dlrurl>  <sms>  <to>95xxxxxxxx</to>  <custom>22</custom>  <custom1>99</custom1>  </sms>  <sms>  <to>99xxxxxxxx</to>  <custom>229</custom>  <custom1>995</custom1>  </sms> </xmlapi>' |

SMS can be sent using the XML API by posting values to the preceding URL by the POST method or the GET method.

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | BASE URL of your SMS Service | URL |
| method | Predefined method | sms.xml |
| xml | You can provide either URL encoded XML data directly or URL of the XML file | Valid XML document or link of XML document |

### Optional Parameter

The Optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/JSON/JSONP. Default JSON |
| callback | Callback function for JSONP response format | Javascript function name standards |

### XML Tag Parameters

***NOTE:*** If message,sender and dlrurl is specified within both SMS tag and XML tag, the former overrides the later.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** | **Type** |
| xml version="1.0 "encoding="UTF-8" | XML Header | Version and Encoding | Mandatory |
| api | Document Root | Document Root | Mandatory |
| flash | Global node for flash | 1/0 | Optional |
| campaign | Global campaign name tag | campaign name | Optional |
| sms | SMS Node(can repeat) | SMS Node | Mandatory |
| to | Mobile number in sms node | Comma separated 10 digit mobile numbers | Required |
| custom | Message ID for sms | comma separated message ID's | Optional |
| custom1 | Custom reference 1 (comma separate) | comma separated custom 1 references | Optional |
| custom2 | Custom reference 2 (comma separate) | comma separated custom 2 references | Optional |
| sender | Sender ID assigned to your account | Sender ID | Mandatory |
| message | Message to be sent. In case of any special characters in your message, enclose it within CDATA a s in the example | Message text | Mandatory |
| dlrurl | The URL to receive delivery reports. Spiffing reference id is mandatory | EX:http://www.example.com/dlr.php | Optional |
| unicode | To specify that the message to be sent is in unicode format.Also can be used for automatic detection of unicode SMS | 1/0/auto | Optional |
| time | Global schedule time | EX Format: YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM | Optional |

**Sample Data*:*** In case of special characters in the content within any of the XML tags, enclose it within CDATA as in following examples.

a. Sending to multiple numbers with same message

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <api>   <campaign>campaign</campaign>   <dlrurl> <![CDATA[http://domain.com/receive?msgid={msgid}&sid={sid}&status={status}&custom1={custom1}]]>  </dlrurl>   <time>2014-12-26 04:00pm</time>   <unicode>0</unicode>   <flash>0</flash>   <sender>senderid</sender>   <message><![CDATA[smstext]]></message>   <sms>   <to>9190\*\*\*\*\*\*\*\*</to>   </sms>   <sms>   <to>9191\*\*\*\*\*\*\*\*</to>   </sms>  </api> |

b. Sending to multiple numbers with different message

|  |
| --- |
| <api>  <campaign>campaign</campaign>  <time>2014-12-26 04:00pm</time>  <unicode>0</unicode>  <flash>0</flash>  <sms>  <to>9190\*\*\*\*\*\*\*\*</to>  <sender>senderid</sender>  <message>smstext</message>  <custom>2</custom>  <dlrurl>  <![CDATA[http://domain.com/receive?msgid={msgid}&sid={sid}&status={status}&custom1={custom1}]]>  </dlrurl>  </sms>  <sms>  <to>9191\*\*\*\*\*\*\*\*</to>  <sender>senderid</sender>  <message><![CDATA[smstext]]></message>  <custom>2</custom>  <dlrurl>  <![CDATA[http://domain.com/receive?msgid={msgid}&sid={sid}&status={status}&custom1={custom1}]]>  </dlrurl>  </sms>  </api> |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "data": [  {  "id": "811436f5-0053-4ea6-8cac-b035880b8473:1",  "customid": "1",  "customid1": "11",  "customid2": "22",  "mobile": "95XXXXXXXXX",  "status": "AWAITED-DLR"  },  {  "id": "811436f5-0053-4ea6-8cac-b035880b8473:2",  "customid": "2",  "customid1": "1",  "customid2": "2",  "mobile": "97XXXXXXXX",  "status": "AWAITED-DLR"  }  ],  "message": "Campaign of 2 numbers Submitted successfully."  } |

# Send an SMS Using JSON API

To send an SMS using JSON API, the following format is used:

|  |
| --- |
| curl -X POST \ '<BASE\_URL>&method=sms.json&json=%7B%22sms%22%3A%20%5B%7B%20%22to%22%3A%20%229xxxxxxxx......%7D%5D%20%7D' |

SMS can be sent using the JSON API by posting values to the preceding URL by the **POST** method with urlencoded json data.

## Sample JSON data

|  |
| --- |
| {  "sms":  [{  "to": "9xxxxxxxx",  "custom": 9xxxxxxxx,  "message": "Message from json api node 1"  },  {  "to": "91xxxxxxxx",  "custom": 34,  "message": "Message from json api node 2"  }]  } |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

## Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | BASE URL of your SMS Service | URL |
| method | Predefined method | sms.json |
| json | You can provide either URL encoded XML data directly or URL of the JSON file | Valid JSON document or link of JSON document |

## Optional Parameter

The Optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |
| callback | Callback function for JSONP response format | Javascript function name standards |

## JSON Tag Parameters

Note: If message, sender and dlrurl is specified within both SMS tag and JSON tag, the former overrides the later.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** | **Type** |
| flash | Global node for flash | 1/0 | Optional |
| sms | SMS Node(can repeat) | SMS Node | Mandatory |
| to | Mobile number in sms node | Comma separated 10 digit mobile numbers | Required |
| custom | Message ID for sms | comma separated message ID's | Optional |
| custom1 | Custom reference 1 (comma separate) | comma separated custom 1 references | Optional |
| custom2 | Custom reference 2 (comma separate) | comma separated custom 2 references | Optional |
| sender | Sender ID assigned to your account | Sender ID | Mandatory |
| message | Message to be sent. In case of any special characters in your message, enclose it within CDATA a s in the example | Message text | Mandatory |
| dlrurl | The URL to receive delivery reports. Spiffing reference id is mandatory | EX: http://www.example.com/dlr.php | Optional |
| unicode | To specify that the message to be sent is in unicode format. Also can be used for automatic detection of unicode SMS | 1/0/auto | Optional |
| time | Global schedule time | EX Format: YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM | Optional |

● Sample Data: In case of special characters in the content within any of the JSON tags, enclose it within CDATA as in following examples.

a. Sending to multiple numbers with same message

|  |
| --- |
| {  "message": "test json",  "sms":[  {  "to": "95XXXXXXXX",  "msgid": "1",  "message": "test sms",  "custom1": "11",  "custom2": "22",  "sender": "AAAAAA"  },  {  "to": "99XXXXXXXX",  "msgid": "2",  "custom1": "1",  "custom2": "2"  }],  "unicode": 1,  "flash": 1,  "dlrurl": "http://www.example.com/dlr.php"  } |

b. Sending to multiple numbers with different message

|  |
| --- |
| {  "message": "test json",  "sms":[  {  "to": "95XXXXXXXX",  "msgid": "1",  "message": "test sms",  "custom1": "11",  "custom2": "22",  "sender": "AAAAAA" },  {  "to": "99XXXXXXXX",  "msgid": "2",  "message": "json test sms",  "custom1": "1",  "custom2": "2"  }],  "unicode": 1,  "flash": 1,  "dlrurl": "http://www.example.com/dlr.php"  } |

## Sample Response

### JSON

|  |
| --- |
| {  "status": "OK",  "data": [  {  "id": "811436f5-0053-4ea6-8cac-b035880b8473:1",  "customid": "1",  "customid1": "11",  "customid2": "22",  "mobile": "95XXXXXXXXX",  "status": "AWAITED-DLR"  },  {  "id": "811436f5-0053-4ea6-8cac-b035880b8473:2",  "customid": "2",  "customid1": "1",  "customid2": "2",  "mobile": "97XXXXXXXX",  "status": "AWAITED-DLR"  }  ],  "message": "Campaign of 2 numbers Submitted successfully."  } |

# Checking SMS Status -PULL

To check status of any sent SMS campaign, you must have message id only (not group ID) of messages sent in the last 3 days. Here is an API for checking the status of an SMS in following format:

|  |
| --- |
| curl -X POST \  '<BASE\_URL>&method=sms.status&id=47xxxxxxxx-x' |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | BASE URL of your SMS Service | URL |
| method | Predefined method | sms.status |
| id | Message ID, comma separated | message id of messages sent in the last 3 days |

### Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/JSON/JSONP. Default response will be in JSON |
| numberinfo | Flag to query service provider and location data | 1 or 0 |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK", "data": [  {  "id": "4429990870:1",  "mobile": "88xxxxxxxx",  "status": "DELIVRD",  "senttime": "2017-02-06 14:54:25",  "dlrtime": "2017-02-06 14:54:26",  "custom": "",  "custom1": null,  "custom2": null,  "provider": "",  "location": ""  } ],  "message": "Processed Successfully"  } |

### Error Response

|  |
| --- |
| {  "status": "A417",  "message": "Invalid Message ID/Mobile. No details found"  } |

### Response Data

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |
| data | All the response is wrapped in side data |
| sms | Each number will be wrapped inside sms |
| id | Website message id for your reference |
| custom1 | Custom message id 1 if you provide one |
| custom2 | Custom message id 2 if you provide one |
| mobile | Mobile number the sms will be sent |
| status | status of the message |
| custom | Custom message id if you provide one |
| senttime | message sent time |
| dlrtime | message delivered time |
| provider | provider of the mobile numer |
| location | Registered location of the mobile number |

# Sending SMS to an Opt In Group

To send an SMS to an opt in group,one must create a optin group and keyword associated with it in your account. Also, need to add numbers for those optin keywords to a group i.e. optin group. An API for sending an SMS to an optin group should be in the following format:

|  |
| --- |
| curl -X POST \  '<BASE\_URL>&method=optin&sender=AAAAAA&message=optin%20test&name=ssss' |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | URL of your SMS Service | URL |
| method | Predefined method | optin |
| sender | Sender ID assigned to your account | Sender ID |
| id/name | id-optin group ID/name-Optin keyword | comma separated id or optin group keyword |
| message | Message to be sent | Message text which is URL encoded(1000 char for normal, 500 for unicode) |

### 

### Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| time | Date and time for scheduling an SMS | EX Format: YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM |
| unicode | To specify that the message to be sent is in unicode format. Also can be used for automatic detection of unicode SMS. | 1 / 0 / auto |
| flash | To specify that the message is to be sent in the flash format | 1 or 0 |
| format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |
| callback | Callback function for JSONP response format | String |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "data": {  "0": {  "id": "4422177667:1",  "customid": "",  "customid1": "",  "customid2": "",  "mobile": "971xxxxxxx",  "status": "AWAITED-DLR"  },  "1": {  "id": "4422177667:2",  "customid": "",  "customid1": "",  "customid2": "",  "mobile": "88xxxxxxx",  "status": "AWAITED-DLR"  },  "2": {  "id": "4422177667:3",  "customid": "",  "customid1": "",  "customid2": "",  "mobile": "95xxxxxxx",  "status": "AWAITED-DLR"  },  "group\_id": 4422177667  },  "message": "Campaign of 3 numbers Submitted successfully."  } |

### Expected Error Codes:

|  |  |
| --- | --- |
| **Status** | **Message** |
| A430 | Group name/id missing |
| A431 | Group is empty |
| A432 | Invalid mobile number |
| A433 | Keyword name/number missing |
| A434 | Keyword not found |
| A435 | Keyword is not active |
| A436 | No sender found! |

### Error Response

|  |
| --- |
| {  "status": "ERROR",  "code": "A431",  "message": "Group is empty"  } |

### Response Data

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your reque  st. |
| message | It indicates the success or failure message. |
| group\_id | Website group id for your reference which is the current date |
| data | All the response is wrapped in side data |
| optin | Each number will be wrapped inside optin group |
| id | Website message id for your reference |
| customid | Custom message id if you provide one |
| customid1 | Custom message id 1 if you provide one |
| customid2 | Custom message id 2 if you provide one |
| mobile | Mobile number the sms will be sent |
| status | status of the message |

# Create a Group

|  |
| --- |
| API to create a group will be in the following format:  curl -X POST \ '<BASE\_URL>&method=groups.add&task=save&app=1' |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | BASE URL of your SMS Service | URL |
| method | Predefined method | groups.add |
| task | Task to be performed for Group | save |
| app | Application reference | 1 |
| data[name] | Defines the name of group | group name |

### Optional Parameter

The Optional parameter in the HTTP API is tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "message": "Group added Successfully",  "data": []  } |

### Error Response

|  |
| --- |
| {  "status": "ERROR",  "message": "Group name already exists.",  "data": []  } |

### Response Data

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |
| data | All the responses are wrapped inside data |

# Add Contacts to a group

First, you need to create a group in your account only then he can add contacts to an existing valid group. API for adding contacts to a group will be in the following format:

|  |
| --- |
| curl -X POST \ '<BASE\_URL>&method=groups.register&number=99xxxxxxxx&name=MyNum&action=add&fullname=xyz&email=xyz%40example.com' |

## API Parameter Specifications

The API parameter specifications are described in the following sections:

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | URL of your SMS Service | URL |
| method | Predefined method | groups.register |
| number | mobile number of the contact (With or without 91) | 99XXXXXXXX or 9199XXXXXXXX |
| name | group name | case insensitive |

### Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| fullname | name of the contact to be added | name of the contact |
| email | email of the contact to be added | email of the contact |
| format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |
| action. Add action is Default | Flag to specify the action | add/delete |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "code": 200,  "message": "Number updated Successfully"  } |

### Error Response:

|  |
| --- |
| {  "status": "ERROR",  "code": "A424",  "message": "Group Name does not exist!"  } |

### API Error Codes

|  |  |
| --- | --- |
| **Status** | **Message** |
| A420 | Group name/id missing |
| A421 | Group is empty/Invalid group |
| A422 | Invalid mobile number |
| A423 | Group name/number missing |
| A424 | Group Name does not exist! |
| A425 | Number does not Exist |

### Response Data

|  |  |
| --- | --- |
| **Status** | **Message** |
| status | It indicates the success or failure of your request. |
| Code | Error code |
| message | It indicates the success or failure message. |

# Send Message to a group

In your account, there must be an existing group and numbers under that group to send any message to a group. API for sending a simple message to a group is in the following format:

|  |
| --- |
| curl -X POST \  '<BASE\_URL>&method=groups&name=MyGroupName&sender=AAAAAA&message=sms+test'  An API for sending a customized message to a group is in the following format:  curl -X POST \  '<BASE\_URL>&method=groups&name=MyGroupName&sender=AAAAAA&message=Dear {name}, Your email is {email}' |

Here, {name} and {email} would be variable parameters and will be picked up from the existing group MYGROUP corresponding to each mobile number. Response formats would be same for simple message to a group or a customized one.

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | URL of your SMS Service | URL |
| method | Predefined method | groups |
| sender | Sender ID assigned to your account | Sender ID (6 alphabets) |
| name | Group name | group name |
| id | Group ID | Comma separated group ID |
| message | Message to be sent | Message text which is URL encoded(1000 char for normal, 500 for unicode) |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "message": "Submitted successfully",  "data": {  "0": {  "id": "4429996893:1",  "customid": "",  "customid1": "",  "customid2": "",  "mobile": "88xxxxxxxxx",  "status": "AWAITED-DLR"  },  "1": {  "id": "4429996893:2",  "customid": "",  "customid1": "",  "customid2": "",  "mobile": "97XXXXXXXX",  "status": "AWAITED-DLR"  },  "group\_id": 4429996893  } } |

### Error Response

|  |
| --- |
| {  "status": "ERROR",  "code": "A421",  "message": "Group is empty/Invalid group" } |

### Response Data

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |
| group\_id | Website group id for your reference |
| data | All the response is wrapped in side data |
| groups | Each number will be wrapped inside groups |
| id | Website message id for your reference |
| customid | Custom message id if you provide one |
| customid1 | Custom message id 1 if you provide one |
| customid2 | Custom message id 2 if you provide one |
| mobile | Mobile number the sms will be sent |
| status | status of the message |

### API Error Codes

|  |  |
| --- | --- |
| **Code** | **Description** |
| A420 | Group name/id missing |
| A421 | Group is empty/Invalid group |
| A422 | Invalid mobile number |
| A423 | Group name/number missing |
| A424 | Group Name does not exist! |
| A425 | Number does not Exist |

# Delete Scheduled SMS

Application must have a scheduled SMS campaign to further modify it and also must save a Group ID of an SMS campaign to be deleted. In order to delete a scheduled slot, there should a minimum gap of 5mins before its execution. API for deleting an SMS campaign that is already scheduled is in the following format:

|  |
| --- |
| curl -X GET \ '<BASE\_URL>&method=sms.schedule&groupid=25xxxxxxxx' |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | URL of your SMS Service | URL |
| method | Predefined method | sms.schedule |
| groupid | group id that has to be deleted | group id that has to be deleted |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "data": "",  "message": "Campaign cancelled successfully and credits are refunded." } |

### Error Response

|  |
| --- |
| {  "status": "A432",  "message": "Campaign already cancelled" } |

### Response Data

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |

# Edit Schedule

Application must have a scheduled SMS campaign to further modify it and must save a Group ID of an SMS campaign to be rescheduled. to edit a scheduled slot, there should a minimum gap of 5mins before its execution. API for editing an SMS slot will be in the following format:

|  |
| --- |
| curl -X GET \ '<BASE\_URL>&method=sms.schedule&groupid=22xxxxxxxx&time=dd%2Fmm%2Fyyyy%20hh%3Amm%20PM&&task=modify' |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| URL | URL of your SMS Service | URL |
| method | Predefined method | sms.schedule |
| groupid | group id whose schedule needs to be edited | group id that has to be deleted |
| task | Specifies that the scheduled time should be modified to the one mentioned within time parameter | modify |
| time | Time to which the slot needs to be re-scheuled | EX Format: YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM |

### Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "data": "",  "message": "Campaign updated successfully" }  Error Response  {  "status":"A431",  "message":"Invalid Group ID" } |

### Response Data

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |

### Expected API Error Codes:

|  |  |
| --- | --- |
| **Code** | **Message** |
| A431 | Invalid Group ID |
| A434 | Campaign within next 5 minutes cannot be modified/cancelled. |

# Check Credits Availability

An API can also be used to check the credits in your our account which can be in the following format:

|  |
| --- |
| curl -X GET \ <BASE\_URL>&method=account.credits' |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| URL | URL of your SMS Service | URL |
| method | Predefined method | account.credits |

### Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "code": "200",  "message": "",  "data":  {  "credits": "-22"  } } |

### Error Response

|  |
| --- |
| {  "status": "A401A",  "message": "Method Not Implemented" } |

### Response Data

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |
| data | All the response is wrapped in side data |
| credits | Displays the credits available in the account |

# HLR Request

To look information about any number, one must put country code along with number. API for performing a lookup for specific numbers is in the following format:

|  |
| --- |
| curl -X POST \ '<BASE\_URL>&method=lookup&to=9192xxxxxxxx' |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| BASE\_URL | URL of your SMS Service | URL |
| method | Predefined method | lookup |
| to | Mobile number along with country prefix to which the SMS is to be sent. Also one provide multiple numbers in comma separated format. | Mobile number(GET- 10numbers,POST- 100numbers) |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "data": [  {  "id": "336-1",  "customid": "",  "customid1": "",  "customid2": "",  "mobile": "97XXXXXXXX",  "status": "AWAITED-DLR"  }],  "message": "Campaign of 1 numbers Submitted successfully." } |

### Error Response

|  |
| --- |
| {  "status":"A400",  "message":"Api Not Implemented" } |

### Response Data:

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |
| data | All the responses are wrapped inside data |
| Mobile | Mobile number |
| id | External message ID |

# Lookup using XML/JSON Data

API for performing a lookup for specific number(s), along with callback URL, will be in the following format:

|  |
| --- |
| curl -X POST \ '<BASE\_URL>&method=lookup.json&to=92xxxxxxxx&json=%3A%2F%2Fdomain.net%2F.......' \  curl -X POST \ '<BASE\_URL>&method=lookup.xml&to=92xxxxxxxx&xml=%3A%2F%2Fdomain.net%2F.......' \ |

## API Parameter Specifications

The API parameter specifications are described in the following sections:

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| SMS\_Service\_URL | URL of your SMS Service | URL |
| method | Predefined method | lookup |
| to | Mobile number along with country prefix to which the SMS is to be sent. Also one provide multiple numbers in comma separated format. | Mobile number(GET- 10 numbers,POST- 100 numbers) |
| xml/json | You can provide either URL encoded XML/JSON data or URL of the XML/JSON file | Valid XML/JSON document or link of XML/JSON document |

### Optional Parameter

The Optional parameter in HTTP API is tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |

### Sample JSON Data

|  |
| --- |
| {  "lookup":[  {  "to": "95XXXXXXXX",  "custom1": "11",  "custom2": "22",  "sender": "AAAAAA"  },  {  "to": "99XXXXXXXX",  "custom1": "1",  "custom2": "2"  }],  "dlrurl": "http://www.example.com/dlr.php/......" } |

### Sample XML Data

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?> <xmlapi>  <dlrurl>  <![CDATA[http://domain.com/dlr.php]]>  </dlrurl>  <lookup>  <to>99xxxxxxxxx</to>  <custom>22</custom>  <custom1>99</custom1>  </lookup>  <lookup>  <to>98xxxxxxxxx</to>  <custom>22</custom>  <custom1>99</custom1>  </lookup> </xmlapi> |

### Custom XML/JSON Tags

|  |  |
| --- | --- |
| **Field Name** | **Description** |
| {orig\_country} | Original Country Name |
| {orig\_circle} | From which circle, number does belong originally |
| {orig\_cprefix} | Original Country Prefix |
| {orig\_nprefix} | Original Network Prefix |
| {orig\_code} | Original geographical circle code |
| {is\_roaming} | Whether number is in roaming or not |
| {roaming\_network} | Roaming Network Name |
| {roaming\_country} | Roaming Country Name |
| {roaming\_circle} | In which circle current number is under roaming |
| {orig\_network} | Original Network Name |
| {roaming\_cprefix} | Roaming Country Prefix |
| {roaming\_nprefix} | Roaming Network Prefix |
| {roaming\_code} | Geographical roaming circle code |
| {is\_ported} | Whether number has been ported or not |
| {ported\_network} | Ported Network Name |
| {ported\_country} | Ported Country Name |
| {ported\_nprefix} | Ported Network Prefix |
| {ported\_code} | Geographical ported circle code |
| {mcc} | Mobile country code used in the combination of mobile network code to uniquely identify an operator |
| {mnc} | Mobile country code used in the combination of mobile network code to uniquely identify an operator |
| {imsi} | 15-digit IMSI number of the mobile subscriber's phone number (MSISDN) |
| {ported\_circle} | To which circle this number has been ported |
| {ported\_cprefix} | Ported Country Prefix |
| {status} | It indicates the success or failure of your request. |
| {smsc} | SMS Centre number |
| {err\_code} | If there is any error or not |

## Example Responses

## JSON

|  |
| --- |
| {  "status": "OK",  "data": [  {  "id": "336-1",  "customid": "",  "customid1": "",  "customid2": "",  "mobile": "97XXXXXXXX",  "status": "AWAITED-DLR"  }],  "message": "Campaign of 1 numbers Submitted successfully." } |

## XML

|  |
| --- |
| <api>  <status>OK</status>  <data>  <lookup>  <id>337-1</id>  <customid></customid>  <customid1></customid1>  <customid2></customid2>  <mobile>97XXXXXXXX</mobile>  <status>AWAITED-DLR</status>  </lookup>  </data>  <message>Campaign of 1 numbers Submitted successfully.</message> </api>  Error Response  <api>  <status>A415</status>  <message>Invalid json</message> </api> |

### Response Data:

|  |  |
| --- | --- |
| **Field Name** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |
| data | All the responses is wrapped inside data |
| Mobile | Mobile number |
| id | External message ID |

## Sample DLRURL

|  |
| --- |
| http://domain.net/dlr/trigger.php?orig\_network={orig\_network}&orig\_country={orig\_country}&orig\_circle={orig\_circle}& orig\_cprefix={orig\_cprefix}&orig\_nprefix={orig\_nprefix}&orig\_code={orig\_code}&is\_roaming={is\_roaming}&roaming\_network={roaming\_network}&roaming\_country={roaming\_country}&roaming\_circle={roaming\_circle}&roaming\_cprefix={roaming\_cprefix}&roaming\_nprefix={roaming\_nprefix}&roaming\_code={roaming\_code}&is\_ported={is\_ported}&ported\_network={ ported\_network}&ported\_country={ported\_country}&ported\_circle={ported\_circle}&ported\_cprefix={ported\_cprefix}&ported\_nprefix={ported\_nprefix}&ported\_code={ported\_code}&mcc={mcc}&mnc={mnc}&imsi={imsi}&status={status}&smsc={smsc}&err\_code={err\_code} |

# Creating a Txtly

|  |
| --- |
| Txtly is basically a shortened URL which can be used in text messages so that SMS would not exceed the characters. API to create a Txtly link is as follows:  curl -X POST \ '<BASE\_URL>&method=txtly.create&url=&token=xyz&title=xyz&advanced=1' |

NOTE: It is mandatory to either provide a URL or attachment that has to be compressed to a short link.

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| URL | URL of your SMS Service | URL |
| method | Predefined method | txtly.create |
| url | URL that requires to be shortened and tracked | Long URL (url\_encoded) |

### Optional Parameters

The Optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| Format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |
| Token | http://msg.mn/heel Here heel is the token. It is unique for each txtly | This can customized word representing your brand/company. If not provided a random unique token is generated |
| Title | A significant title to your txtly | If not provided, your txtly will not contain any title |
| Advanced | advanced analytics gives an option to track who (Recipient mobile numbers) visited the page | 1- will enable advanced analytics/0(default) - will disable advanced analytics |
| Track | Location Track gives the city and state details of URL visitor | 1- will enable location tracking/0(default) - will disable location tracking |
| Attach | media file that requires to be compressed to a short link. | Provide the media file in a CURL request |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "message": "Details saved Successfully",  "token": "5yx1",  "id":"33527",  "txtly": "http://msg.mn/bKX6yf"  } |

### Error Response

|  |
| --- |
| {  "status": "E500",  "message": "Please provide url to redirect" } |

## Response Data

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |
| token | Returns the unique token of that txtly |
| txtly | compressed url that is ready to use |
| id | ID of the txtly created |

## Expected Error Codes

|  |  |
| --- | --- |
| **Code** | **Description** |
| E500 | Please provide url to redirect |
| E501 | Txtly Already Exists. Please try another one. |
| E502 | Please attach a file |
| E503 | Invalid file format. Upload valid file. |
| E504 | File size exceed. Maximum 50000 Kb. |
| E505 | File upload failed. Please try again. |

# Deleting a Txtly

One must have a txtly id to delete it from database. Here is an API to delete a Txtly link:

|  |
| --- |
| curl -X GET \ '<BASE\_URL>&method=txtly&task=delete&id=31xx&app=1' |

# API Parameter Specifications

The API parameter specifications are described in the following sections.

## Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| URL | URL of your SMS Service | URL |
| method | Predefined method | txtly |
| task | Task to be performed on txtly | delete |
| id | id of the txtly | 123 |
| app | Application reference | 1 |

## Optional Parameters

The Optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "message": "Deleted successfully..", "data": [ ]  } |

### Error Response

|  |
| --- |
| {  "status": "ERROR",  "message": "Id entered not exits with this api user", "data": []  } |

### Response Data

|  |  |
| --- | --- |
| **Field** | **Description** |
| status | It indicates the success or failure of your request. |
| message | It indicates the success or failure message. |
| data | All the responses is wrapped inside data |

# Txtly Reports Extraction - PULL

API to Pull logs for all txtly shortened links in your account:

|  |
| --- |
| curl -X GET \  '<BASE\_URL>&method=txtly&app=1' |

## API Parameter Specifications

The API parameter specifications are described in the following sections

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| URL | URL of your SMS Service | URL |
| method | Predefined method | txtly |
| app | Txtly Application Logs | 1 |

### Optional Parameter

The Optional parameter in the HTTP API is tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| format | Output format should be as specified by this variable | XML/PHP/JSON/JSONP. Default response will be in JSON |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "message": "OK",  "data":  {  "txtlys":  [{  "link\_id": "426272",  "title": "",  "token": "Eu6uVN",  "long\_url": "https://www.google.co.in/search?q=url.  &oq=so&aqs=chrome.1.69i60j69i59j69i60l3j69i59.  1358j0j7&sourceid=chrome&ie=UTF-8",  "advanced": "0",  "link\_type": "URL",  "callback": "",  "meta\_value": "{\"track\":null}",  "views": "1",  "last\_viewed": "1486730380",  "status": "1",  "created": "1486730303",  "serial": 1,  "short\_url": "http://msg.mn/Eu6uVN"  {  "link\_id": "426068",  "title": "",  "token": "wNO6c2",  "long\_url": "https://www.google.co.in/search?q=url.  &oq=solut&aqs=chrome.1.69i60j69i59j69i60l3j69i57.  2228j0j7&sourceid=chrome&ie=UTF-8",  "advanced": "0",  "link\_type":"URL",  "callback": "",  "meta\_value": "{\"track\":null}",  "views": "0",  "last\_viewed": "0",  "status": "1",  "created": "1486725578",  "serial": 2,  "short\_url": "http://msg.mn/wNO6c2"  },  {  "link\_id": "426067",  "title": "",  "token": "KcIaHc",  "long\_url": "https://www.google.co.in/search?  q=url.&oq=solut&aqs=chrome.  1.69i60j69i59j69i60l3j69i57.2228j0j7  &sourceid=chrome&ie=UTF-8",  "advanced": "0",  "link\_type": "URL",  "callback": "",  "meta\_value": "{\"track\":null}",  "views": "0",  "last\_viewed": "0",  "status": "1",  "created": "1486725490",  "serial": 3,  "short\_url": "http://msg.mn/KcIaHc"  }  }],  "pagination":  {  "now": 3,  "next": false,  "page": 1,  "limit": 25,  "limitstart": 0,  "total": 3,  "extra": null  }  }  } |

# Pull logs for Individual Txtly

Here is an API to pull logs for individual txtly from our account:

|  |
| --- |
| curl -X POST \ '<BASE\_URL>&method=txtly.logs&id=5xxxxx&app=1' |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Expected Values** |
| URL | URL of your SMS Service | URL |
| method | Predefined method | txtly.logs |
| app | Txtly Application Logs | 1 |
| id | Id of the txtly for which you want logs | 123 |

## Example Responses

### JSON

|  |
| --- |
| {  "status": "OK",  "message": "OK",  "data":  {  "txtlys":  [{  "log\_id": "11219199",  "fkuserid": "7458555458",  "fk\_link\_id": "426272",  "client\_ip": "240520452894295621769",  "scheme": "",  "host": "",  "query\_string": "",  "user\_agent": "Mozilla/5.0  (Linux; Android 7.0; Moto G(4)  Build/NPJ25.93-14)AppleWebKit/537.36  (KHTML,like Gecko)  Chrome/55.0.2883.91 Mobile Safari/537.36",  "browser": "Chrome Mobile",  "browser\_version": "55.0.2883.",  "browser\_lang": "en",  "browser\_engine": "Blink",  "resolution": "",  "platform": "Android",  "platform\_version": "7.0",  "device\_type": "smartphone",  "device\_brand": "Motorola",  "device\_version": "",  "device\_model": "",  "touch\_enabled": "0",  "latitude": "",  "longitude": "",  "country": "",  "region": "",  "city": "",  "status": "1",  "created": "10-Feb 06:09 PM",  "mobile": null,  "serial": 1  }],  "pagination":  {  "now": 1,  "next": false,  "page": 1,  "limit": 25,  "limitstart": 0,  "total": 1,  "extra": null  }  }  } |

# Send an Email

Through Email API, you may send email to multiple contacts, schedule emails and track analytics of such emails. You may use any of the following email content parameters while executing email campaigns:

* Text - plain text content to be sent
* Html - full html content to be sent
* Template – pre-defined templated to be used

## Send Simple Text Email

|  |
| --- |
| curl -X POST \  '<BASE\_URL>&method=email&format=xml&subject=testingemails&to[0][email]=<emailID>&to[0][name]=ABC&from[email]=<emailID>&text=Hi how are you today&from[name]=XYZ' |

## API Parameter Specifications

The API parameter specifications are described in the following sections.

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| URL | URL | URL of Alerts service |
| api\_key | Alphanumeric characters (API key) | Authentication key shared with you from SI |
| method | Email or List | email to send an email to single or multiple recipient. list to send an email to everyone in your contact group |
| text | body of email | Plain text message to be displayed in email |
| subject | Subject of email | Subject to be displayed to recipient in email |
| From[email] | Sender email ID | From which email ID you want to send email |
| To[0][email] | Recipient email ID; Can also be used for multiple email IDs in sequence like To[1][email], To[2][email], etc | To which email ID(s) you want to send email(s) |

### Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| To[0][name] | Name of recipient to be shown in email; Can also be used for multiple recipients associated with email IDs like To[1][name], to[2][name],etc | To whom you want to send email(s); Recipient name(s) |
| From[name] | Who has sent this email(s) i.e. Sender Name to be shown in email | Sender name so that recipients must know that you have sent email(s) |
| format | XML/PHP/JSON/JSONP. | In which format you want to see the response. Default response will be in JSON |
| time | YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM | Date and time need to be mentioned; when you want to schedule an email. |
| name | Campaign name | You may give any name to this email campaign |

## Example Responses

### JSON

|  |
| --- |
| { "status": "OK",  "data": {  "0": {  "id": "1878054-1",  "email": "xyz@gmail.com",  "name": ABC,  "status": "AWAITED-DLR"  },  "group\_id": 1878054  },  "message": "Campaign of 1 emails Submitted successfully."} |

### XML

|  |
| --- |
| <api>  <status>OK</status>  <data>  <group\_id>1878063</group\_id>  <email>  <id>1878063-1</id>  <email>xyz@gmail.com</email>  <name>abc</name>  <status>AWAITED-DLR</status>  </email>  </data>  <message>Campaign of 1 emails Submitted successfully.</message>  </api> |

* If you have put so many email IDs in a single parameter, then system will pick the last email ID. For eg. To[0][email]= recipient\_one@example.com , To[0][email]= recipient\_two@example.com To[0][email]= recipient\_three@example.com To[0][email]= recipient\_four@example.com, then in this case, system will send an email to recipient\_four@example.com.
* We need to put email IDs in the following format & sequence only: To[0][email]= recipient\_one@example.com , To[1][email]= recipient\_two@example.com To[2][email]= recipient\_three@example.com To[3][email]= recipient\_four@example.com
* Parameters will be be mapped in the following way while sending emails: To[0][email]= recipient\_one@example.com & to[0][name]=Recipient One To[1][email]= recipient\_two@example.com & to[1][name]=Recipient Two

## Send An Email With Attachment

|  |
| --- |
| curl -X POST \  '<BASE\_URL>&method=email&format=xml&subject=testingemails&to[0][email]=<emailID>&to[0][name]=ABC&from[email]=<emailID>&text=Hi how are you today&from[name]=XYZ&files[]=sdfrt12.png' |

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| URL | URL | URL of Alerts service |
| api\_key | Alphanumeric characters (API key) | Authentication key shared with you from SI |
| method | Email | To send an email to single and multiple recipients as well |
| text | body of email | Plain text message to be displayed in email |
| subject | Subject of email | Subject to be displayed to recipient in email |
| From[email] | Sender email ID | From which email ID you want to send email |
| To[0][email] | Recipient email ID; Can also be used for multiple email IDs in sequence like To[1][email], To[2][email], etc | To which email ID(s) you want to send email(s) |

### Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| Files[] | Attachment upto 10Mb | Attach any file to be sent to recipients; Recipients will receive emails along with the attachments upto 10Mb |
| To[0][name] | Name of recipient to be shown in email; Can also be used for multiple recipients associated with email IDs like To[1][name], to[2][name],etc | To whom you want to send email(s); Recipient name(s) |
| From[name] | Who has sent this email(s) i.e. Sender Name to be shown in email | Sender name so that recipients must know that you have sent email(s) |
| format | XML/PHP/JSON/JSONP. | In which format you want to see the response. Default response will be in JSON |
| time | YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM | Date and time need to be mentioned; when you want to schedule an email. |
| name | Campaign name | You may give any name to this email campaign |

# Send an Email using Template

Here is an API format to send email using the pre-defined templates created using “Template” feature under Emailer menu. If you use any specific templates, then you may skip Text or HTML parameter i.e. Body of an email as system will pick the template content for body of an email.

|  |
| --- |
| curl -X POST \ '<BASE\_URL>&method=email&format=xml&subject=testingemails&to[0][email]=<emailID>&to[0][name]=ABC&from[email]=<emailID&from[name]=XYZ&files[]=sdfrt12.png&template[id]=30' |

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| URL | URL | URL of Alerts service |
| api\_key | Alphanumeric characters (API key) | Authentication key shared with you from SI |
| method | Email | To send an email to single and multiple recipients as well |
| subject | Subject of email | Subject to be displayed to recipient in email |
| From[email] | Sender email ID | From which email ID you want to send email |
| To[0][email] | Recipient email ID; Can also be used for multiple email IDs in sequence like To[1][email], To[2][email], etc | To which email ID(s) you want to send email(s) |

### Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| Template[id] | Template ID (numeric value) for specific template | Use this template ID for pre-defined templates created in the application; template will be applied to outgoing email campaign |
| Files[] | Attachment upto 10Mb | Attach any file to be sent to recipients; Recipients will receive emails along with the attachments upto 10Mb |
| To[0][name] | Name of recipient to be shown in email; Can also be used for multiple recipients associated with email IDs like To[1][name], to[2][name],etc | To whom you want to send email(s); Recipient name(s) |
| From[name] | Who has sent this email(s) i.e. Sender Name to be shown in email | Sender name so that recipients must know that you have sent email(s) |
| format | XML/PHP/JSON/JSONP. | In which format you want to see the response. Default response will be in JSON |
| time | YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM | Date and time need to be mentioned; when you want to schedule an email. |
| name | Campaign name | You may give any name to this email campaign |

# Send an HTML email

You can apply CSS styles to such HTML content based emails and your outgoing mail will adopt these styles across all recipients.

|  |
| --- |
| curl -X POST \ '<BASE\_URL>&method=email&format=xml&subject=testingemails&to[0][email]=<emailID>&to[0][name]=ABC&from[email]=<emailID> &from[name]=XYZ&files[]=sdfrt12.png&template[id]=30&html= <div class="form-group"> <label for="password">Password</label> <input type="password" name="password" class="form-control" required="required" placeholder="Enter your password" autocomplete="off" /> <a href="http://domain\_URL/members/resetpass" id="forgot-password">Forgot?</a> </div>' |

### Mandatory Parameters

The mandatory parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| URL | URL | URL of Alerts service |
| api\_key | Alphanumeric characters (API key) | Authentication key shared with you from SI |
| method | Email | To send an email to single and multiple recipients as well |
| Html (Mandatory for customized emails) | Html tags needs to be used for designing emails | To include design elements in email content via HTML tags |
| subject | Subject of email | Subject to be displayed to recipient in email |
| From[email] | Sender email ID | From which email ID you want to send email |
| To[0][email] | Recipient email ID; Can also be used for multiple email IDs in sequence like To[1][email], To[2][email], etc | To which email ID(s) you want to send email(s) |

### Optional Parameters

The optional parameters in the HTTP API are tabulated below:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| Template [id] or | Template ID (numeric value) for specific template | Use this template ID for pre-defined templates created in the application; template will be applied to outgoing email campaign |
| Files[] | Attachment upto 10Mb | Attach any file to be sent to recipients; Recipients will receive emails along with the attachments upto 10Mb |
| To[0][name] | Name of recipient to be shown in email; Can also be used for multiple recipients associated with email IDs like To[1][name], to[2][name],etc | To whom you want to send email(s); Recipient name(s) |
| From[name] | Who has sent this email(s) i.e. Sender Name to be shown in email | Sender name so that recipients must know that you have sent email(s) |
| format | XML/PHP/JSON/JSONP. | In which format you want to see the response. Default response will be in JSON |
| time | YYYY-MM-DD HH:MM:SS OR YYYY-MM-DD HH:MM AM/PM | Date and time need to be mentioned; when you want to schedule an email. |
| name | Campaign name | You may give any name to this email campaign |

### Expected API Error Codes

|  |  |
| --- | --- |
| **STATUS** | **MESSAGE** |
| E601 | Missing recipient |
| E602 | Invalid Template |
| E603 | Missing message |
| E604 | Invalid schedule time. Minimum 5minutes time gap is required. |
| E606 | Unable to find gateway to send email |
| E607 | No valid email address found |
| E609 | Unable to write to temp files |
| E610 | Unable to process your request. try again |
| E599 | Missing From address |
| A401B | Method not found |
| E600 | Missing Subject |
| A402 | API key not found |

# Developers Guide

## PHP

|  |
| --- |
| <?php  include 'Sendsms.php';  // Pass api key and senderid of your account $sendsms = new Sendsms("https://domain\_URL", "apikeyXXXXX", "XXXXXX");  $dlr\_url = 'http://www.yourdomainname.domain/yourdlrpage&custom=XX';  // Sending an sms instantly  $sendsms->sendSms("99XXXXXXXX", "message", $dlr\_url);  // Schedule an sms params are mobile, message, dlrurl, format, time $sendsms->scheduleSms("99XXXXXXXX", "message", 'YYYY-MM-DD HH:MM PM/AM', $dlr\_url);  //Sending an unicode sms params are mobile, message, dlr url, format , 1 for unicode $sendsms->unicodeSms("99XXXXXXXX", "unicode message", $dlr\_url);  // To know delivery status of mobile number $sendsms->msgdlvStatus("messageid");  //$sendsms->setWorkingKey("1i6xxxxxxxxxxxxxx"); //$sendsms->setSenderId("BUxxxx"); //$sendsms->setApiurl("https://domain\_URL"); ?> |

## JAVA

|  |
| --- |
| public class call  {  public static void main(String[] args) throws Exception  {  SendSms smsObj = new SendSms();  /\*   \* Setting up parameters  \* Method can be sms or sms.status  \* Api\_key which you can get from developer section  \* Sender\_Id from which you want to execute the SMS campaign  \*/  smsObj.setParams("https://domain\_URL", "sms", "Api\_key", "Sender\_Id");  /\*  \* Sending Normal SMS  \* Mobile\_Number: valid mobile number with country prefix  \* Text\_Message: SMS content  \* Dlr\_Url: Delivery url ex : http://www.yourdomainname.domain/yourdlrpage&custom=XX  \*/  smsObj.sendSms("Mobile\_Number;", "Text\_Message", "Dlr\_Url");  //smsObj.scheduleSms("Mobile\_Number", "Text\_Message", "Dlr\_Url", "YYYY-MM-DD HH:MM:SS");  //smsObj.sendUnicodeSms("Mobile\_Number", "Text\_Message", "Dlr\_Url");  //smsObj.scheduleUnicodeSms("Mobile\_Number", "Text\_Message", "Dlr\_Url", "YYYY-MM-DD HH:MM:SS");  //smsObj.messageDeliveryStatus("Msg\_Id");  //smsObj.groupDeliveryStatus("Group\_Id");  //smsObj.setSenderId("Sender\_Id");  //smsObj.setApiKey("Api\_Key");  //smsObj.setApiUrl("URL");  }  } |

## ASP.NET

|  |
| --- |
| Imports System Imports System.Net Imports System.IO Imports System.Web Dim sURL As String sURL = "https://domain\_URL/v4/?method=sms&api\_key=A62XXXXXXXXXXXXXXX&to=95XXXXXXX&sender=XXXXXX&message=test" + HttpUtility.UrlEncode("This is test message") Dim wrGETURL As WebRequest wrGETURL = WebRequest.Create(sURL) Dim objStream As Stream objStream = wrGETURL.GetResponse.GetResponseStream() Dim objReader As New StreamReader(objStream) Dim sLine As String = "" Dim i As Integer = 0 sLine = objReader.ReadLine If Not sLine Is Nothing Then MsgBox(sLine) End If |