

Lertify, Sending SMS via API

Datum: 1.4.2026

Overview

The Lertify SMS API allows sending SMS messages to one or multiple recipients via a simple HTTP interface.

Endpoint

```
POST https://api.lertify.app/v1/sms/messages
```

Authentication

Requests must include an API key in the HTTP header.

Header	Type	Required	Description
apikey	string	Yes	API key used for authentication

Request

Content-Type: application/json

Body Parameters

Field	Type	Required	Description
sender	string	Yes	Sender name or phone number displayed to recipient
destinations	array[string]	Yes	List of recipient phone numbers
message.content	string	Yes	SMS message content
message.isUnicode	boolean	No	Indicates whether the message uses Unicode
deliveryReportUrl	string	No	URL where delivery reports will be sent

Example Request

```
{
  "sender": "Lertify01",
  "destinations": [
    "+38640123456",
    "+38640987654"
  ],
  "message": {
    "content": "Test SMS message",
    "isUnicode": false
  },
  "deliveryReportUrl": "https://example.com/delivery-report"
}
```

Response

Status code

Code	Description
200	Request accepted for processing

Response Structure

Field	Type	Description
messages	array	List of sent messages
messages[].messageId	UUID	Unique message identifier
messages[].destination	string	Recipient phone number
status	string	Request status

Example Response

```
{
  "messages": [
    {
      "messageId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
      "destination": "+38640123456"
    }
  ],
  "status": "accepted"
}
```

Notes

- The endpoint supports sending multiple SMS messages in a single request
- Phone numbers must be in international format with a + prefix (e.g. +38640123456)
- deliveryReportUrl enables asynchronous delivery status notifications

Delivery report (Webhook)

Overview

If deliveryReportUrl is provided in the request, the system will send HTTP POST requests to the specified URL with information about SMS delivery status.

Endpoint Requirements

- Must accept **HTTP POST** requests
- Must return **HTTP 2xx status**

If a non-2xx response is returned, the retry mechanism is triggered.

Retry Mechanism

If the endpoint does not return a successful response (HTTP 2xx), the system retries delivery using the following form:

```
delay = (poskus2) + 1 sekund
```

- Maximum number of attempts: 10
- Retry intervals increase progressively

Retry Schedule

Attempt	Delay	Total elapsed time
1	2 min	2 min
2	5 min	7 min
3	10 min	17 min
4	17 min	34 min
5	26 min	60 min (1 h)
6	37 min	97 min (1 h 37 min)
7	50 min	147 min (2 h 27 min)
8	65 min (1 h 5 min)	212 min (3 h 32 min)
9	82 min (1 h 22 min)	294 min (4 h 54 min)
10	101 min (1 h 41 min)	395 min (6 h 35 min)

Example Webhook Payload

```
{
  "messageId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "status": "DELIVERED"
}
```

Notes

- Delivery reports are sent per recipient

Code Examples

PHP

```
<?php

$url = "https://api.lertify.app/v1/sms/messages";
$apiKey = "YOUR_API_KEY";

$data = [
  "sender" => "Lertify01",
  "destinations" => [
    "+38640123456"
  ],
  "message" => [
    "content" => "Test SMS message",
    "isUnicode" => false
  ],
  "deliveryReportUrl" => "https://example.com/delivery-report"
];

$ch = curl_init($url);

curl_setopt($ch, CURLOPT_HTTPHEADER, [
  "Content-Type: application/json",
  "apikey: " . $apiKey
]);

curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($data));

$response = curl_exec($ch);

if (curl_errno($ch)) {
  echo 'Error: ' . curl_error($ch);
} else {
  echo "Response: " . $response;
}

curl_close($ch);
```

Python

```
import requests

url = "https://api.lertify.app/v1/sms/messages"

headers = {
    "Content-Type": "application/json",
    "apikey": "YOUR_API_KEY"
}

data = {
    "sender": "Lertify01",
    "destinations": [
        "+38640123456",
        "+38640987654"
    ],
    "message": {
        "content": "Test SMS message",
        "isUnicode": False
    },
    "deliveryReportUrl": "https://example.com/delivery-report"
}

response = requests.post(url, json=data, headers=headers)

print("Status code:", response.status_code)
print("Response:", response.json())
```