

VIBER API

Auth:

Basic Auth

Example:

```
$userHashKey = 'User Hash Key provided by your account manager';  
$ch = curl_init('https://web.it-decision.com/v1/api/send-viber');  
curl_setopt($ch, CURLOPT_RETURNTRANSFER,1);  
curl_setopt($ch, CURLOPT_HTTPAUTH, CURLAUTH_BASIC);  
curl_setopt($ch, CURLOPT_USERPWD, "$userHashKey");  
curl_setopt($ch, CURLOPT_TIMEOUT, 30);  
curl_setopt($ch, CURLOPT_POST, 1);  
curl_setopt($ch, CURLOPT_POSTFIELDS, $requestParams); // $requestParams  
- request array with correct data  
$result = curl_exec($ch);  
curl_close($ch);
```

API Send Viber message:

POST: <https://web.it-decision.com/v1/api/send-viber>

Request POST:

```
{  
    "source_addr": "Custom Company",  
    "destination_addr": 8882222200,  
    "message_type":106,  
    "text":"Message content",  
  
    "image":"https://yourdomain.com/images/image.jpg",  
    "button_caption":"Join Us",  
    "button_action":"https://yourdomain.com/join-us",  
    "source_type":1,  
    "callback_url":"https://yourdomain.com/viber-callback",  
    "validity_period":3600  
}
```

source_addr:

<= 20 chars - from whom the message

destination_addr:

<= 20 chars - to whom the message

message_type (Type of message sent):

106 only text (convenient for transactional messages)

108 text+image+button (convenient for promotional messages)

206 only text (2Way)*(convenient for promotional messages)

208 text+image+button (2Way)* (convenient for promotional messages)

text:

<= 1000 chars - text of Viber message

image (Correct URL with image for promotional message with button caption and button action):

jpg or jpeg (mime type is image/jpeg), maximum resolution 400x400 pixels

png (mime type is image/png), maximum resolution 400x400 pixels

button_caption:

<= 30 chars - button caption

button_action:

Correct URL for transition when you press the button

source_type (Message sending procedure):

promotion message (the message can be with text, picture and button) - 1

transactional message (text message) - 2

callback_url:

Correct URL for message status callback

validity_period:

TTL (Time to live) allows the sender to limit the life time of a message. In case the message did not get the status "delivered" before the time ended, the message will not be charged and will not be delivered to the user. In case no TTL was provided (no "ttl" parameter) Viber will try to deliver the message for up to 1 day.

promotion message - min TTL 40 seconds max TTL 21600 seconds (6 hours)

transactional message - min TTL 40 seconds max TTL 21600 seconds (6 hours)

Response JSON (POST):

```
{  
  "message_id":429  
}
```

message_id:

Sent message ID

* Get message from a user for 2Way messages:

For 2way messages ITD Viber system will send callbacks with each user's message. The tracking data content will be sent by the client according to the tracking data in the last message which was received on the Viber client's side.

The response implies that the API user has a callback url for this messages.

Response JSON (POST):

```
{
  "message_token": 4444444444444444,
  "phone_number": "97251222222",
  "time": 2121212121,
  "message": {
    "text": "a message to the service",
    "tracking_data": "tracking_id:100035"
  }
}
```

message_token:

client response message token

phone_number:

client phone number

time:

client response message time

message:

text: client response message text

tracking_data:

tracking_id: The id of the message to which the client is responding

API Receive Viber message:

POST: <https://web.it-decision.com/v1/api/receive-viber>

Request POST:

```
{
  "message_id":429
}
```

message_id:

The ID of the message whose status you want to get (for the last 5 days)

Response JSON:

```
{  
  "message_id":429,  
  "status":1,  
}
```

message_id:

The ID of the message whose status you want to get (for the last 5 days)

status:

Current Viber message status

Viber messages statuses:

sent - 0;
delivered - 1;
error - 2;
rejected - 3;
undelivered - 4;
pending - 5;
unknown - 20;

Errors:

name : Too Many Requests
message : Rate limit exceeded.
code : 0
status : 429

name : Invalid Parameter: [param_name],
message : Empty parameter or parameter validation error,
code : 1,
status : 400,

name : Internal server error,
message : The server encountered an unexpected condition which prevented it
from fulfilling the request,
code : 2,
status : 500,

name : Topup balance is required,
message : User balance is empty,
code : 3,
status : 402,