



Integración REST Text2Speech

Versión 1.1

Índice de Contenidos

Introducción	Pág. 3
Plataforma Técnica	
Petición de envíos de mensajes de voz	Pág. 4
Petición JSON	Pág. 4
Ejemplo de petición CURL	Pág. 5
Ejemplo de petición PHP	Pág. 5
Códigos de estado de respuestas	Pág. 6
Política de reintentos	Pág. 7

Introducción

La plataforma REST Gateway permite al usuario enviar mensajes de voz a través de HTTP o HTTPS de una manera sencilla y rápida, pudiendo enviar más de 500 mensajes en una sola petición. Para poder acceder a sus estadísticas y datos de facturación puede acceder a la web <http://www.nrsgateway.com/> con sus datos de usuario.

Esta documentación describe los parámetros necesarios para utilizar todas las posibilidades para el envío de mensajes de voz siguiendo las especificaciones REST. Tanto las peticiones como las respuestas de la API REST están en formato JSON, haciendo muy sencillo la utilización de la API con cualquier lenguaje de programación.

PLATAFORMA TÉCNICA

Petición de envío de mensaje de voz

Cada petición que se realice tendrá que incluir en la cabecera de la petición http la autenticación del cliente. Para ello se utiliza la autenticación de acceso básica de HTTP.

La cabecera de autorización se construye combinando la cadena “usuario: contraseña” y codificándola en base64. A esta cadena se antepone la cadena “Authorization: Basic”

Por ejemplo, para el usuario “miuser” y el password “mipass” la cabecera resultante sería:

```
Authorization: Basic bWl1c2VyOm1pcGFzcw==
```

A continuación se detallará las opciones de envío disponibles, la URL a la que se debe llamar, y los parámetros que admite.

Para generar la URL el cliente deberá hacer una llamada POST a la siguiente dirección:

<https://gateway.plusmms.net/voice/message>

Petición JSON:

Ejemplo de petición básica:

```
{"to":["346666666666"],"text":"Esto es un test de mensaje de voz","gender":"F","language":"es_ES"}
```

Parámetros requeridos:

- **text:** texto del mensaje. Como máximo puede tener 500 caracteres. El texto tiene que estar codificado en UTF-8
- **to:** número de teléfono móvil destinatario del mensaje. Debe incluir el prefijo (Ej: En España 34666666666). Este campo permite indicar múltiples destinatarios, para ello hay incluir todos los números en un array.
- **language:** Lenguaje que se quiere utilizar para convertir el texto, las opciones son:
'en_GB' => 'Ingles - Reino Unido',
'en_US' => 'Ingles - Estados unidos',
'es_CO' => 'Español - Latino',
'es_ES' => 'Español - España',
'pt_BR' => 'Portugués - Brasil'
- **gender:** Genero de la voz, los valores permitidos son 'F' para voz de mujer y 'M' para voz de hombre.

Parámetros opcionales:

- **valPeriod:** Periodo de validez. Permite determinar la fecha máxima en la que la llamada se puede realizar. Transcurrida esta fecha límite, la llamada será cancelada. Especificando el periodo de validez se activará automáticamente los reintentos de llamada, que volverá a realizar la llamada si el destinatario no atiende la llamada. La fecha debe tener el formato YYYYmmddHHiiSS (Ej: 20130215142000 sería el 15 de febrero de 2013 a las 14:20)

Ejemplo de petición CURL:

```
curl -X POST \  
-H "Content-Type: application/json" \  
-H "Accept: application/json" \  
-H "Authorization: Basic bWl1c2VyOm1pcGFzcw==" \  
-d "{\"to\":[\"346666666666\"],\"text\": \"mensaje de  
texto\", \"language\": \"es_ES\", \"gender\": \"M\"}" \  
https://gateway.plusmms.net/voice/message
```

Ejemplo de petición PHP

```
<?php  
$post['to'] = array('34666555444');  
$post['text'] = "mensaje de texto";  
$post['gender'] = 'F';  
$post['language'] = 'es_ES';  
  
$user = "miuser";  
$password = 'mipass';  
  
$ch = curl_init();  
curl_setopt($ch, CURLOPT_URL, "https://gateway.plusmms.net/voice/message");  
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);  
curl_setopt($ch, CURLOPT_POST, 1);  
curl_setopt($ch, CURLOPT_HEADER, 1);  
curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($post));  
curl_setopt($ch, CURLOPT_HTTPHEADER, array(  
    "Accept: application/json",  
    "Authorization: Basic ".base64_encode($user.":".$password)  
));  
  
$result = curl_exec ($ch);  
?>
```

La clave de acceso (password) y el código del cliente (username) serán proporcionados por la empresa. Hay que comentar que con objeto de aumentar la seguridad del sistema, el cliente deberá indicar la IP desde donde se va a conectar, solo se permitirán envíos de la IP indicada por el cliente.

Códigos de estado de respuestas

La API REST puede responder con los estados HTTP siguientes:

Código estado	Descripción	Detalles
202	Accepted	El mensaje se ha aceptado para su posterior proceso
207	Multi-status	El mensaje se ha aceptado para su posterior proceso, pero algunos de los destinatarios son incorrectos.
400	Bad request	La petición contiene errores, el mensaje no ha sido aceptado
401	Unauthorized	Fallo en la autenticación del cliente
402	Payment required	El cliente no dispone de saldo suficiente
500	Internal server error	El servidor ha tenido un error interno

En el cuerpo de la respuesta HTTP se entrega un JSON con los detalles del resultado, estas son las respuestas posibles:

Código estado 202:

```
[{"accepted":true,"to":"34666555444","id":"102648819"}]
```

Código estado 207:

```
[{"accepted":true,"to":"34626690739","id":"102648820"},  
{"accepted":false,"to":"34","error":{"code":102,"description":"No valid recipients"}}]
```

Código estado 400:

```
{"error":{"code":102,"description":"No valid recipients"}}  
{"error":{"code":104,"description":"Text message missing"}}  
{"error":{"code":105,"description":"Text message too long"}}  
{"error":{"code":113,"description":"Invalid coding"}}  
{"error":{"code":114,"description":"Invalid voice gender"}}  
{"error":{"code":115,"description":"Invalid language"}}  
{"error":{"code":116,"description":"Invalid validity period"}}]
```

Código estado 401:

```
{"error":{"code":103,"description":"Username or password unknown"}}  
{"error":{"code":112,"description":"IP address not allowed"}}]
```

Código estado 402:

```
{"error":{"code":111,"description":"Not enough credits"}}]
```

Política de reintentos

El sistema Text2Speech permite reintentar automáticamente las llamadas si el destinatario no atiende la llamada. Esta funcionalidad no genera costes añadidos, ya que solo se cobrarán las llamadas que sean atendidas por el destinatario.

Cuando el sistema detecta que la llamada no ha sido atendida, se encola nuevamente pasados 30 minutos. El tiempo transcurrido desde la llamada no atendida hasta el reintento será variable dependiendo de la cola existente, pero siempre superior a 30 minutos. Se reintentará un máximo de 3 veces después de la primera llamada.

Para evitar que los reintentos se realicen en horario no deseados es imprescindible especificar la variable opcional **valPeriod** para que esta funcionalidad se active.