

REST API – SMS Gateway

Waltool ofrece una API y herramientas para integrar su servicio de consulta de información y mensajería SMS con desarrollos de terceros.

Con este sistema, usted podrá enviar mensajes directos o responder consultas SMS desde sus sistemas.

Este documento, contiene la información básica para utilizar la API de Waltool.

Junto con este documento, encontrará una serie de archivos y carpetas que componen el Kit de Desarrollo (SDK), con funciones que facilitan la implementación de este sistema y ejemplos prácticos.

El funcionamiento de este sistema, utilización, compatibilidad, costos y términos de uso podrán cambiar sin previo aviso.

Tenga en cuenta que la API se encuentra en fase de desarrollo, se podrán experimentar fallas durante su aplicación. El equipo de Waltool posee un canal de comunicación abierto para reportar inconvenientes o dejar sus comentarios y sugerencias: <http://www.waltool.com/soporte>

Funcionamiento

El servidor de Waltook recibe los mensajes que se generan desde un sistema externo para enviarlos a teléfonos móviles (y viceversa).

La comunicación entre Waltook y el sistema externo se realiza vía consultas HTTP.



Por cuestiones de seguridad, las consultas se encriptan mediante un algoritmo propio que se asocia a un ID de cliente (CLIENT_ID) y una clave de seguridad (CLIENT_KEY).

Cuando el sistema desee *enviar un mensaje a un teléfono*, generará una consulta HTTP a Waltook con los parámetros del mensaje a enviar.



En cambio, cuando se trate de un mensaje *recibido desde un teléfono*, será Waltook quien se contacte con el sistema notificando la recepción del mensaje y adjuntando los datos del mismo (Método 1: Callback).



Además, el sistema externo podrá realizar una consulta a Waltook para obtener la lista de mensajes recibidos leídos o pendientes de lectura (Método 2).



En el caso de que el método 1 falle por problemas de conectividad o del sistema externo, Waltook volverá a intentar conectarse con el servidor periódicamente y tendrá disponible el mensaje para ser consultado a través del método 2.

Herramientas

Waltool provee las herramientas para facilitar el desarrollo de aplicaciones gracias a su kit para desarrolladores (Waltool SDK). Actualmente sólo disponible para PHP.

De todas formas, se pondrá a disposición el algoritmo de encriptación y funciones de ayuda para que se puedan adaptar a otros lenguajes y plataformas.

Las funciones incluidas en el SDK facilitan las tareas de comunicación entre su sistema y Waltool. Contienen los algoritmos de encriptación, parseo y resolución de errores.

SDK (Kit de desarrollo)

En el Kit de Desarrollo, encontrará los siguientes archivos:

Archivos del motor de Waltool:

config.php

Archivo de configuración (requerido siempre por waltool_connect.php)

waltool_connect.php

Funciones para utilizar la API (requerido siempre)

Archivos con ejemplos de uso:

enviar.php- Demostración de un envío

recibir.php - Demostración de recepción (callback)

consultar.php - Obtener el listado de mensajes en formato serializado.

consultar_xml.php - Obtener el listado anterior pero en XML

consultar_json.php - Idem anteriores pero en JSON.

credito.php - Consulta crédito de mensajes de entrada y salida disponibles

Configurando el SDK

Datos de su cuenta

Para utilizar los scripts incluidos en el SDK, primero deberá acceder a su panel de servicios Waltook y dirigirse a la sección API, o bien clickeando en el siguiente enlace: <http://www.waltook.com/panel/api>

Allí encontrará los datos de configuración de su API. Observará los valores CLIENT_ID y CLIENT_KEY. Abra el archivo config.php que encontrará en el SDK e ingrese en las variables los datos que encontrará en el panel.

Por ejemplo:

```
define('WALTOOK_API_CLIENT_ID',118);  
define('WALTOOK_API_CLIENT_KEY','82d91b1b0a00ec41d98d18b');
```

Métodos de envío de datos

Por default, los scripts del SDK se conectan con el servidor de Waltook a través del método POST. Pero si tiene algún inconveniente, podrá cambiar al método GET modificando la constante WALTOOK_API_METHOD:

```
define('WALTOOK_API_METHOD','GET');
```

Envío de mensajes

Para responder a una consulta o enviar un mensaje directo a un usuario, su sistema deberá realizar una consulta HTTP a una URL de Waltook enviando su CLIENT_ID y los datos del mensaje, ya encriptados.

La función `waltook_send_message` simplifica esta tarea.

Por favor, revise el archivo **enviar.php** que encontrará en el Kit de Desarrollo donde se explica como enviar un mensaje con esta función.

Sintaxis

```
$status=waltook_send_message($message);
```

En `$status` se almacenará el estado de la transacción (INTEGER). Ver **apartado sobre códigos de error y estado**.

\$message corresponde a un array con la siguiente estructura:

```
array (  
    'uid' => '1123456789',  
    'txt' => 'Respuesta a su consulta',  
    'tid' => 554562,  
)
```

Información de los parámetros

UID: Teléfono(s). *Requerido.*

Ingresar el número de teléfono **sin** el código de país, **con** el código de área, **sin** cero, **sin** quince, **sólo caracteres numéricos**. Para múltiples destinatarios, separar con comas.

Por ejemplo:

1151991777 **Válido**

+541151991777 **No válido**

11-5199-1777 **No válido**

01151991777 **No válido**

1551991777 **No válido**

011 15 5199 1777 **No válido**

TXT: Mensaje que envió (no incluye el código principal que identifica su servicio). *Requerido. Máximo 160 caracteres.*

TID: Corresponde al identificador del mensaje original (que originó la consulta). *Opcional. Para más información, vea más adelante el apartado “Recepción de mensaje”.*

Ejemplo de uso de la función de envío

Mensaje único

```
<?php

//Funciones API Waltook
require("waltook_connect.php");

//Parámetros del mensaje a enviar
$w_data['uid']="1151991777"; //Reemplace por su núm de teléfono
$w_data['txt']="Testeando"; //Reemplace por el mensaje
$w_data['tid']=0;

//Envía el mensaje y guarda el estado en la variable $status
$status=waltok_send_message($w_data);

//Toma el código de estado y verifica si se envió (Cod. 1) o si se
generó un error (cualquier otro valor distinto de 1)
if($status==1)
{
    print "Mensaje enviado correctamente";
}else
{
    print "Error al enviar el mensaje. Cód: ".$status;
}

?>
```

Múltiples destinatarios de un mismo mensaje

```
<?php

//Funciones API Waltook
require("waltook_connect.php");

//Parámetros del mensaje a enviar
$w_data['uid']="1151991777,1153131415,1125121241";
$w_data['txt']="Testeando"; //Reemplace por el mensaje
$w_data['tid']=0;

//Envía el mensaje y guarda el estado en la variable $status
$status=waltok_send_message($w_data);

//Toma el código de estado y verifica si se envió (Cod. 1) o si se
generó un error (cualquier otro valor distinto de 1)
if($status==1)
{
    print "Mensaje enviado correctamente";
}else
{
    print "Error al enviar el mensaje. Cód: ".$status;
}

?>
```

Recepción de mensajes

En su servidor deberá contar con una URL donde obtendrá los datos correspondientes a cada mensaje que reciba (método 1: Callback) o bien, consultar a Waltook sus mensajes pendientes de lectura (método 2).

Método 1: Callback

Waltook iniciará una consulta GET a esa URL cada vez que llegue un mensaje, enviando los datos de manera encriptada.

Cuando su script esté listo, deberá ingresar en su panel de servicios la URL del script al cual Waltook enviará los datos cada vez que se reciba un mensaje.

En el kit de desarrollo (SDK) encontrará el archivo **recibir.php** con el ejemplo de aplicación.

La función **waltook_capture_message** tomará la información que envíe Waltook, la desencriptará y la pondrá a su disposición entregando un array con la siguiente estructura:

```
array (  
    'uid' => '1123456789',  
    'txt' => 'Ejemplo de consulta',  
    'timestamp' => 1320801583,  
    'tid' => 554562,  
    'status' => 1  
)
```

Información de los valores de respuesta

UID: Teléfono del usuario

TXT: Mensaje que envió (no incluye el código principal que identifica su servicio)

TIMESTAMP: Hora y fecha en que se recibió

TID: ID de la consulta

STATUS: Resultado de la transacción (Vea el **apartado sobre códigos de estado y error**)

Deberá responder con un mensaje de confirmación para que el servidor de Waltook considere que este mensaje fue enviado correctamente. Para ello, encontrará dos funciones que realizan esta tarea (**waltook_reply_ok** o **waltook_reply_error**).

Ejemplo

```
<?php

//Funciones API Waltook
require("waltook_connect.php");

//Toma la variable que envió Waltook, la descripta, la
procesa y separa los datos del mensaje almacenándolos en
un array.
$wd=waltook_capture_message();
```

```
//Si el array contiene el texto del mensaje, lo imprime
if($wd['txt'])
{
    $mensaje="Mensaje de: ".$wd['uid']."\n";
    $mensaje.=$wd['txt'];

    $archivo =
fopen("recibidos/".date("Y_m_d_H_i_s").".txt","w");
    fwrite($archivo,$mensaje);
    fclose($archivo);

    //Imprime el mensaje de confirmación
    waltook_reply_ok();
}else
{

    //imprime el mensaje de error, waltook considerará
que el mensaje no está leído y volverá a intentar
enviarlo más tarde.
    waltook_reply_error();
}

?>
```

IMPORTANTE

- No deberá procesar con este script el mensaje sino almacenarlo para su posterior procesamiento y respuesta. Los tiempos de demora en el procesamiento del mensaje dentro de su sistema podrían entorpecer la comunicación entre ambos servidores y la transacción podría considerarse fracasada.
- La API rechazará cualquier salida que no corresponda a la confirmación de recepción o aviso de error. El script sólo podrá responder llamando a las funciones `waltook_reply_ok()` y `waltook_reply_error()`. No deberá tener ningún otro carácter en la salida (tampoco espacios). Chequee sus etiquetas de apertura y cierre de PHP para evitar que no haya caracteres adicionales.
- Si falla la recepción del mensaje, Waltook volverá a intentar luego (hasta 3 veces). Si la conexión continúa fallando luego de 3 intentos, volverá a intentar cuando se reciba un nuevo mensaje. De todas formas, podrá recuperarlo utilizando el método 2.

Método 2

Para consultar a Waltook sobre sus mensajes pendientes de lectura (o ya leídos), deberá llamar a una URL de Waltook (adjuntando su CLIENT_ID y el TID del último mensaje que recibió correctamente) y obtendrá el listado encriptado en formato XML, serializado o JSON.

En el SDK se incluyen dos funciones que facilitan esta tarea (`waltook_get_messages` y `waltook_get_messages_array`) entregando los mensajes en distintos formatos junto a una variable de estado.

Con el método 2, puede procesar los mensajes en el momento con el mismo script y enviar las respuestas a Waltook, ya que no habría demoras que puedan estropear las consultas.

En el archivo **consultar.php** encontrará un ejemplo de aplicación de la función **waltook_get_messages_array()**.

Sintaxis

```
$miarray=waltook_get_messages_array($msgstatus, $tid,$flag);
```

Parámetros

\$msgstatus

Estado de los mensajes a leer. *Opcional*

Valores:

1: Mensajes leídos

0: Mensajes no leídos

null: Todos los mensajes (default)

\$tid

Corresponde al ID (TID) a partir del cual se mostrarán los mensajes (utilizado para cuando se desea consultar los mensajes pendientes desde la última recepción de cierto mensaje). Predeterminado: 0. *Opcional*

\$flag

Si se desea que Waltook considere que todos los mensajes que estaban pendientes de lectura y acaban de ser entregados pasen al estado “Leído” (0) o se mantengan en el estado “No leídos” (1). Ideal para utilizar en fases de desarrollo. Predeterminado: 1. *Opcional*

La función entregará un array con la siguiente estructura:

```
array (  
  'status' => 1,  
  'messages' =>  
    array (  
      0 =>  
        array (  
          'uid' => '1123456789',  
          'txt' => 'Esta es la primera consulta',  
          'timestamp' => 1320871028,  
          'tid' => 554562,  
        ),  
      1 =>  
        array (  
          'uid' => '1198765432',  
          'txt' => 'Esta es la segunda consulta',  
          'timestamp' => 1320871028,  
          'tid' => 554563,  
        ),  
    ),  
)
```


Información de los valores de respuesta

UID: Teléfono del usuario

TXT: Mensaje que envió (no incluye el código principal que identifica su servicio)

TIMESTAMP: Hora y fecha en que se recibió

TID: ID de la consulta

STATUS: Refleja el estado de la consulta - Vea el **apartado sobre códigos de estado y error**

Ejemplo

```
<?php

//Funciones API Waltook
require("waltook_connect.php");

//Seleccionar todos los mensajes (leídos y no leídos) y no
cambiar el estado de los no leídos
$a=waltok_get_messages_array(null,0,0);

//Si hay más de un mensaje, mostrarlos en la tabla
if(count($a['messages'])>0)
{
    $estados=array(0=>"No leído",1=>"Leído");
    print "<table>";
    print
    "<tr><td>Teléfono</td><td>Mensaje</td><td>Timestamp</td><td>
Estado</td></tr>";
```

```
        foreach($a['messages'] as $m)
        {
            print
            "<tr><td>".$m['uid']. "</td><td>".$m['txt']. "</td><td>".$m['t
            imestamp']. "</td><td>".$estados[$m['status']]. "</td></tr>";

        }

        print "</table>";
    }else
    {
        print "No hay mensajes";
    }

?>
```

Si desea obtener directamente la respuesta en formato SERIALIZADO, XML o JSON, utilice la función **waltook_get_messages**.

Sintaxis

```
$datos=waltook_get_messages($msgstatus, $tid,$format,$flag);
```

Parámetros

\$msgstatus

Estado de los mensajes a leer.

Valores: 1: Mensajes leídos ... 0: Mensajes no leídos ... *null*: Todos los mensajes. Predeterminado: *null*. *Opcional*

\$tid

Corresponde al ID (TID) a partir del cual se mostrarán los mensajes (utilizado para cuando se desea consultar los mensajes pendientes desde la última recepción de cierto mensaje). Predeterminado: 0. *Opcional*

\$flag

Si se desea que Waltook considere que todos los mensajes que estaban pendientes de lectura y acaban de ser entregados pasen al estado “Leído” (0) o se mantengan en el estado “No leídos” (1). Ideal para utilizar en fases de desarrollo. Predeterminado: 1. *Opcional*

\$formato

Indica cómo se entregará la información:

1 = Serializado

2 = JSON

3 = XML

Predeterminado: 1. *Opcional*

Ejemplo

```
<?php

//Seleccionar todos los mensajes (leídos y no leídos,
desde el primero, en JSON y sin cambiar el estado de los
no leídos

$a=waltook_get_messages(null,0,2,0);

print $a;

?>
```

Todos los formatos entregarán dos elementos padre (messages y status).

Status

Refleja el estado de la consulta

Vea el **apartado sobre códigos de estado y error**

Messages

Contiene los mensajes de la consulta

Registro de mensajes

Waltok le provee un registro interactivo de mensajes enviados y recibidos con buscador y tracking de mensajes.

Se encuentra en su panel de servicios (<http://www.waltok.com/panel/api>) con el nombre “Log de entrada” y “Log de salida”.

Consulta de crédito

Puede consultar a través de la API cuántos mensajes le quedan disponibles para enviar y recibir. También puede observar esos datos en su panel de servicios (<http://www.waltok.com/panel/api>).

En el SDK, encontrará el script `credito.php` con el ejemplo de consulta.

Hay dos funciones disponibles para consultar su crédito:

`waltok_get_credit()` y `waltok_get_credit_array()`.

waltok_get_credit()

Devuelve el crédito en formato serializado, JSON o XML.

Sintaxis

```
$datos= waltok_get_credit($format);
```

El parámetro `$format` corresponde a 1:Serializado, 2:JSON y 3:XML

waltook_get_credit()

Devuelve el crédito en formato serializado, JSON o XML.

Sintaxis

```
$datos= waltook_get_credit_array();
```

Devolverá un array donde \$datos['credit'] contendrá los valores 'in' y 'out'.

Ejemplo

```
<?php
require("waltook_connect.php");

$data=waltook_get_credit_array();

if($data['credit']['out'])
{
    print "Entrada: ".$data['credit']['in']."<br>";
    print "Salida: ".$data['credit']['out'];
}else
{
    print "Error al enviar el mensaje. Cód:
".$data['status'];
}
?>
```

Códigos de estado y error

Las funciones de Waltook podrán devolver los siguientes códigos según el resultado de cada transacción.

1	OK - Transacción exitosa
2	Error de conexión con api.waltook.com
3	Error de lectura. No se pudo descryptar.
4	Teléfono no válido (fijo o inexistente)
5	No se recibieron datos.
7	Faltan parámetros (UID: Teléfono y TXT: Mensaje)
8	Mensaje original (TID) no válido
9	Número de teléfono no válido (UID) - Sólo números de Argentina, con prefijo DDN sin 0 y sin 15.
10	Mensaje muy largo (TXT) - Máximo 160 caracteres
11	Cuenta suspendida
12	No hay crédito suficiente para realizar la operación

Soporte

Recuerde que el equipo de Waltook posee un canal de comunicación abierto para brindarle ayuda, recibir reportes de inconvenientes o dejar sus comentarios y sugerencias:

<http://www.waltook.com/soporte>