

# Envío y recepción de mensajería

## WhatsApp

Manual Técnico v6.2.2

### DECLARACIÓN DE CONFIDENCIALIDAD

Este documento es confidencial y privilegiado; está protegido contra la divulgación y puede incluir información de propiedad de Concepto Móvil®. Concepto Móvil® proporciona este documento con el entendido de que se mantendrá en la más estricta confidencialidad y no será divulgado, reproducido o utilizado, en todo o parte, para ningún fin que no sea el alcance de la sociedad de cambio aquí documentada.

© Concepto Móvil 2019. Todos los derechos reservados.

**Historial del documento**

Fecha	Versión	Descripción	Autor
03 de Junio 2019	1.0	Creación del documento	Ing. Juan del Angel Pérez
04 de Junio 2019	2.0	Ajustes en formato y redacción	Ing. Pedro Rivera Luna
29 de Noviembre 2019	3.0	Acotación de información	Ing. Pedro Rivera Luna
20 de enero 2021	6.2.2	Nuevo administrador de colas	Ing. Alexis Sánchez Bustamante

**Revisores**

Fecha	Versión	Descripción	Autor
03 de Enero 2020	5.0	Revisar contenido del documento	Ing. Gabriel Gutiérrez
03 de Enero 2020	6.0	Modificación de contenido del documento	Ing. Alejandro Villarroel
07 de Enero 2020	6.1	Cambios menores en el documento	Ing. Alejandro Villarroel
14 de Enero 2020	6.2	Correcciones mínimas	Ing. Alejandro Villarroel
06 Febrero 2020	6.2.1	Se agregó el flujo de envío de mensajes	Ing. Gabriel Gutiérrez
21 de enero 2011	6.2.2	Se revisa la información del administrador de colas	Ing. Yessica Lucía García Moreno

## Contenido

1. Propósito .....	4
2. Flujo para el intercambio de mensajes .....	4
3. Variables.....	4
4. Contactos .....	5
4.1. Alta de contactos.....	5
4.2. Baja de contactos .....	8
5. Mensajes .....	11
5.1. Envío de mensajes.....	11
5.2. Consumo de API por Whitelist .....	16
6. Callbacks.....	17
6.1. Recepción de mensajes .....	17
6.1.1. Mensajes entrantes .....	18
6.1.2. Informes de entrega.....	20
6.2. Seguridad .....	21
6.3. Manejo de colas .....	21
6.3.1. Recepción de Notificaciones de WhatsApp.....	22
6.3.2. Procesamiento asíncrono de Notificaciones de WhatsApp .....	22
6.3.3. Procesamiento asíncrono de notificaciones fallidas .....	24

## 1. Propósito

El propósito de este documento es proporcionar la información técnica necesaria que ayude a los técnicos encargados de la configuración de envío y recepción de mensajería WhatsApp consumiendo la API de Concepto Móvil®.

El manual proporciona la información técnica para el envío de texto, imágenes, audio, archivos, geolocalización y videos por medio de WhatsApp.

## 2. Flujo para el intercambio de mensajes

- Antes de mandar cualquier mensaje el contacto debe estar registrado.
- Solo se puede iniciar una conversación con un template.
- Una vez reciba respuesta del cliente comienza una sesión de 24 horas, durante las cuales podrá enviar al cliente mensajes.
- Cada que el cliente responde comienza un nuevo ciclo de 24 horas.
- Si una sesión caduca deberá reiniciar con un template.
- Si el cliente manda mensaje sin haber iniciado una conversación, se creará una sesión de 24 horas, pero para tener intercambio de mensajes con el cliente asegúrese de registrarlo primero.

## 3. Variables

Estas variables sirven como referencia en los endpoints de este documento de la API de Broadcasterbot.

**{host}**: <https://whatsappapi.broadcasterbot.com/v1>

**{apiKey}**: Token de autorización proporcionado por Concepto Móvil

El token de autorización es un Bearer token, esto quiero decir que el token debe ir concatenado con un “Bearer ” al principio.

## 4. Contactos

### 4.1. Alta de contactos

Todas las conversaciones iniciadas por el negocio deben iniciar con la “alta del contacto”.

Request	Descripción			
<b>Url</b>	{host}/v1/contacts/register/			
<b>Method</b>	POST			
<b>Headers</b>	Nombre	Tipo	Descripción	Valor
	Authorization	String	Llave de autorización	Bearer {apiKey}
	Content-Type	String		application/json
<b>Query Params</b>	N/A			
<b>Path Params</b>	N/A			
<b>Body</b>	<b>Ejemplo:</b>			
	<pre>{   "contacts": [     {       "phone": "5544332211",       "country": "MX"     }   ] }</pre>			
	<b>Atributos: (* Requerido)</b>			
	Nombre	Tipo	Descripción	
	* contacts	Object	Arreglo de objetos	
	* contacts[*].phone	String	Número local	
	* contacts[*].country	String	Código del país del número del contacto (según ISO 3166 Alfa-2)	

Response	Descripción																									
<b>Headers</b>	N/A																									
<b>Body</b>	<b>Ejemplo:</b> <pre>{   "httpStatus": 0,   "message": "string",   "data": [     {       "createdAt": "2020-01-03T19:06:06.352Z",       "updatedAt": "2020-01-03T19:06:06.352Z",       "phone": "5544332211",       "country": "MX"     }   ] }</pre>																									
	<b>Atributos:</b> <table> <tr> <th>Nombre</th><th>Tipo</th><th>Descripción</th></tr> <tr> <td>httpStatus</td><td>Integer</td><td>Estatus Http de la respuesta</td></tr> <tr> <td>message</td><td>String</td><td>Mensaje de la respuesta</td></tr> <tr> <td>data</td><td>Object</td><td>Lista de contactos dados de alta</td></tr> <tr> <td>data[*].createdAt</td><td>String</td><td>Fecha de creación (ISO)</td></tr> <tr> <td>data[*].updatedAt</td><td>String</td><td>Fecha de actualización (ISO)</td></tr> <tr> <td>data[*].phone</td><td>String</td><td>Número local</td></tr> <tr> <td>data[*].country</td><td>String</td><td>Código del país del número del contacto (según ISO 3166 Alfa-2)</td></tr> </table>		Nombre	Tipo	Descripción	httpStatus	Integer	Estatus Http de la respuesta	message	String	Mensaje de la respuesta	data	Object	Lista de contactos dados de alta	data[*].createdAt	String	Fecha de creación (ISO)	data[*].updatedAt	String	Fecha de actualización (ISO)	data[*].phone	String	Número local	data[*].country	String	Código del país del número del contacto (según ISO 3166 Alfa-2)
Nombre	Tipo	Descripción																								
httpStatus	Integer	Estatus Http de la respuesta																								
message	String	Mensaje de la respuesta																								
data	Object	Lista de contactos dados de alta																								
data[*].createdAt	String	Fecha de creación (ISO)																								
data[*].updatedAt	String	Fecha de actualización (ISO)																								
data[*].phone	String	Número local																								
data[*].country	String	Código del país del número del contacto (según ISO 3166 Alfa-2)																								

Response	Descripción		
<b>Error Body</b>	<b>Ejemplo:</b>		
	<pre>{   "httpStatus": 0,   "message": "string",   "code": 0,   "errors": {} }</pre>		
	<b>Atributos:</b>		
	Nombre	Tipo	Descripción
	httpStatus	Integer	Estatus Http de la respuesta
	message	String	Mensaje de la respuesta
	code	Number	Codigo de error (listados en la tabla de errores)
	errors	Object	Descripción de errores
	<b>Errores:</b>		
	HttpStatus	Code	Descripción
	400 (Bad Request)	1300	El cuerpo contiene una petición con sintaxis errónea (Valores inválidos, campos faltantes, etc.)
	401 (Unauthorized)	1100	ApiKey inválida
	401 (Unauthorized)	1101	No se proporcionó ApiKey

## 4.2. Baja de contactos

Dar de baja contactos previamente dados de alta.

Request	Descripción			
<b>Url</b>	{host}/v1/contacts/delete/			
<b>Method</b>	POST			
<b>Headers</b>	Nombre	Tipo	Descripción	Valor
	Authorization	String	Llave de autorización	Bearer {apiKey}
	Content-Type	String		application/json
<b>Query Params</b>	N/A			
<b>Path Params</b>	N/A			
<b>Body</b>	<b>Ejemplo:</b>			
	<pre>{   "contacts": [     {       "phone": "5544332211",       "country": "MX"     }   ] }</pre>			
	<b>Atributos: (* Requerido)</b>			
	Nombre	Tipo	Descripción	
	* contacts	Object	Arreglo de objetos	
	* contacts[*].phone	String	Número local	
	* contacts[*].country	String	Código del país del número del contacto (según ISO 3166 Alfa-2)	



Response	Descripción		
<b>Headers</b>	N/A		
<b>Body</b>	<b>HttpStatus: 200 (Ok)</b>		
	<b>Ejemplo:</b>		
	<pre>{   "httpStatus": 0,   "message": "string",   "data": null }</pre>		
	<b>Atributos:</b>		
	Nombre	Tipo	Descripción
	httpStatus	Integer	Estatus Http de la respuesta
<b>Error Body</b>	message	String	Mensaje de la respuesta
	data	Object	Respuesta vacía
	<b>Ejemplo:</b>		
	<pre>{   "httpStatus": 0,   "message": "string",   "code": 0,   "errors": {} }</pre>		
	<b>Atributos:</b>		
	Nombre	Tipo	Descripción
	httpStatus	Integer	Estatus Http de la respuesta
	message	String	Mensaje de la respuesta
	code	Number	Codigo de error (listados en la tabla de errores)
	errors	Object	Descripción de errores

Response	Descripción		
	<b>Errores:</b>		
	HttpStatus	Code	Descripción
	400 (Bad Request)	1301	El cuerpo contiene una petición con sintaxis errónea (Valores inválidos, campos faltantes, etc.)
	401 (Unauthorized)	1100	ApiKey inválida
	401 (Unauthorized)	1101	No se proporcionó ApiKey

## 5. Mensajes

### 5.1. Envío de mensajes

Este endpoint es el que se utilizará para el envío de mensajes a cualquier número que tenga WhatsApp.

Solo se puede enviar un campo a la vez (“media”, “location”, “text”, “template”). Los otros campos que no se envíen pueden tener un valor `null` o simplemente no venir en el cuerpo de la petición. En caso de que se llegue a mandar mas de un campo de los mencionados, solo se tomara en cuenta uno.

Request	Descripción			
Url	{host}/v1/messages/			
Method	POST			
Headers	Nombre	Tipo	Descripción	Valor
	Authorization	String	Llave de autorización	Bearer {apiKey}
	Content-Type	String		application/json
Query Params	N/A			
Path Params	N/A			
Body	<b>Ejemplos:</b> <b>Enviar un template</b> <pre>{  "contacts": [    {      "phone": "5544332211",      "country": "MX"    },    {      "phone": "5511223344",      "country": "MX"    }  ],  "template": {    "name": "my_template_name",    "params": [      "John",      "Doe"    ]  }}</pre>			

Request	Descripción
	<pre> ], "language": "es_mx" "ttl": "P1D"  } } </pre>
	<b>Enviar un texto</b> <pre> {   "contacts": [     {       "phone": "5544332211",       "country": "MX"     },     {       "phone": "5511223344",       "country": "MX"     }   ],   "text": {     "body": "Hello world!",     "previewUrl": false   } } </pre>
	<b>Enviar un archivo multimedia (Imagen, Video, Audio, Documento)</b> <pre> {   "contacts": [     {       "phone": "5544332211",       "country": "MX"     },     {       "phone": "5511223344",       "country": "MX"     }   ],   "media": {     "type": "image",     "url": "http://exampledomain/file1.pdf",     "caption": "This is an example file caption",     "filename": "myimage.png"   } } </pre>

Request	Descripción																											
	<div><div>Enviar una ubicación</div><pre>{  "contacts": [    {      "phone": "5544332211",      "country": "MX"    },    {      "phone": "5511223344",      "country": "MX"    }  ],  "location": {    "latitud": "19.4259214",    "longitud": "-99.1556818",    "placeName": "El Ángel de la Independencia",    "address": "Paseo de la Reforma y, Eje 2 Pte, Juárez, Cuauhtémoc, 06600 Ciudad de México, CDMX"  } }</pre></div> <div>Atributos:</div> <table><tr><th>Nombre</th><th>Tipo</th><th>Descripción</th></tr><tr><td>* contacts</td><td>Object</td><td>Arreglo de objetos</td></tr><tr><td>* contacts[*].phone</td><td>String</td><td>Número local</td></tr><tr><td>* contacts[*].country</td><td>String</td><td>Código del país del número del contacto (según ISO 3166 Alfa-2)</td></tr><tr><td>template</td><td>Object</td><td>Objeto con parámetros para envío de template</td></tr><tr><td>* template.name</td><td>String</td><td>Nombre del template a enviar</td></tr><tr><td>* template.params</td><td>Array&lt;String&gt;</td><td>Parámetros del template</td></tr><tr><td>* template.language</td><td>String</td><td>Idioma del template (Ejemplo: en, en_us, es_mx)</td></tr><tr><td>text</td><td>Object</td><td>Objeto con parámetros para enviar un texto</td></tr></table>	Nombre	Tipo	Descripción	* contacts	Object	Arreglo de objetos	* contacts[*].phone	String	Número local	* contacts[*].country	String	Código del país del número del contacto (según ISO 3166 Alfa-2)	template	Object	Objeto con parámetros para envío de template	* template.name	String	Nombre del template a enviar	* template.params	Array<String>	Parámetros del template	* template.language	String	Idioma del template (Ejemplo: en, en_us, es_mx)	text	Object	Objeto con parámetros para enviar un texto
Nombre	Tipo	Descripción																										
* contacts	Object	Arreglo de objetos																										
* contacts[*].phone	String	Número local																										
* contacts[*].country	String	Código del país del número del contacto (según ISO 3166 Alfa-2)																										
template	Object	Objeto con parámetros para envío de template																										
* template.name	String	Nombre del template a enviar																										
* template.params	Array<String>	Parámetros del template																										
* template.language	String	Idioma del template (Ejemplo: en, en_us, es_mx)																										
text	Object	Objeto con parámetros para enviar un texto																										

Request	Descripción		
	* text.body	String	Contenido del mensaje de texto
	* text.previewUrl	Boolean	Objeto de mensaje
	media	Object	Objeto con parámetros para enviar un archivo multimedia
	* media.type	String	Tipo de archivo (“image”, “video”, “audio”, “document”)
	* media.url	String	URL pública del archivo. Debe ser un enlace HTTP o HTTPS
	media.caption	String	Texto opcional que será anidado junto con el archivo
	* media.filename	String	Nombre del archivo
	* media.mimeType	String	Tipo del archivo (Ejemplo: application/pdf)
	location	Object	Objeto con parámetros para enviar una ubicación
	* location.latitud	String	Latitud
	* location.longitud	String	Longitud
	location.placeName	String	Nombre del lugar. Sera desplegado en el mensaje
	location.address	String	Dirección. Sera desplegada en el mensaje

Response	Descripción		
Headers	N/A		
Body	<b>HttpStatus: 201 (Created)</b>		
	<b>Ejemplo:</b>		
	<pre>{   "httpStatus": 201,   "data": [     {       "status": "success",       "state": "delivered",       "messageId": "fe93211f-9f28-426f-853d-6f85b3f53be6",       "recipient": {         "phone": "5544332211",         "country": "MX"       }     }   ] }</pre>		
	<b>Atributos:</b>		
	Nombre	Tipo	Descripción
	httpStatus	Integer	Estatus Http de la respuesta
message	String	Mensaje de la respuesta	
data	Array<Object>	Lista de los estatus de los mensajes enviados. Lleva un formato similar a los informes de entrega. ( <a href="#">Ver informes de entrega</a> )	
Error Body	<b>Ejemplo:</b>		
	<pre>{   "httpStatus": 0,   "message": "string",   "code": 0,   "errors": {} }</pre>		
	<b>Atributos:</b>		
	Nombre	Tipo	Descripción
	httpStatus	Integer	Estatus Http de la respuesta
	message	String	Mensaje de la respuesta

Response	Descripción		
	code	Number	Código de error (listados en la tabla de errores)
	errors	Object	Descripción de errores
	<b>Errores:</b>		
	HttpStatus	Code	Descripción
	400 (Bad Request)	1200	El cuerpo contiene una petición con sintaxis errónea (Valores inválidos, campos faltantes, etc.)
	401 (Unauthorized)	1100	ApiKey inválida
	401 (Unauthorized)	1101	No se proporcionó ApiKey

## 5.2. Consumo de API por Whitelist

Para consumir el API de BroadcasterBot, se debe solicitar al equipo de operaciones la habilitación de la IP del servidor que consumirá el servicio para que el equipo de Concepto Móvil agregue la dirección a la whitelist de BroadcasterBot.

No es posible habilitar el consumo del API de BroadcasterBot sin este esquema de seguridad.



## 6. Callbacks

Un callback es una petición POST HTTP con una notificación hecha por la API de WhatsApp de Broadcasterbot la URL de tu elección. La API de WhatsApp espera que el servidor receptor envíe una respuesta 2xx (Exitosa).

### 6.1. Recepción de mensajes

Un callback siempre contendrá alguna de las siguientes estructura:

```
{  
  "messages": []  
}
```

O

```
{  
  "statuses": []  
}
```

Nombre	Tipo	Descripción
messages	Array<Object>	Arreglo de mensajes entrantes
statuses	Array<Object>	Arreglo de informes de entrega

Es importante considerar que la respuesta 2xx de este servicio no debe depender de un servicio externo, ni siquiera del envío de mensajes por la API de BroadcasterBot, únicamente de la recepción correcta de las peticiones de messages y/o statuses. La lógica de negocio debe manejarse de manera independiente.

### 6.1.1. Mensajes entrantes

Un mensaje entrante o MO (Mobile originated) es un mensaje enviado por un usuario de Whatsapp.

Cada mensaje puede contener los campos “text”, “media” o “location” dependiendo del tipo de mensaje, pero no puede contener los 3 campos al mismo tiempo.

```
{
  "messages": [
    {
      "messageId": "fe93211f-9f28-426f-853d-6f85b3f53be6",
      "contact": {
        "phone": "5544332211",
        "country": "MX"
      },
      "text": {
        "body": "Hello world!",
        "previewUrl": false
      },
      "media": {
        "type": "document",
        "url": "http://exampledomain/file1.pdf",
        "caption": "This is an example file caption",
        "filename": "myfile.pdf",
        "mimeType": "application/pdf"
      },
      "location": {
        "latitud": "19.4259214",
        "longitud": "-99.1556818",
        "placeName": "El Ángel de la Independencia",
        "address": "Paseo de la Reforma y, Eje 2 Pte, Juárez, Cuauhtémoc, 06600 Ciudad de México, CDMX"
      }
    }
  ]
}
```

Nombre	Tipo	Descripción
messageId	String	Id único del mensaje. Con este campo se hace el informe de entrega ( <a href="#">Informes de entrega</a> )
contact	Object	Arreglo del contacto que manda el mensaje
contacts.phone	String	Número local
contacts.country	String	Código del país del número del contacto (según ISO 3166 Alfa-2)

text	Object	Objeto de mensaje de text
text.body	String	Contenido del mensaje de texto
text.previewUrl	Boolean	Objeto de mensaje
media	Object	Objeto de mensaje de archivo multimedia
media.type	String	Tipo de archivo (“image”, “video”, “audio”, “document”)
media.url	String	URL pública del archivo. Debe ser un enlace HTTP o HTTPS
media.caption	String	Texto opcional que será anidado junto con el archivo
media.filename	String	Nombre del archivo
media.mimeType	String	Tipo del archivo (Ejemplo: application/pdf)
location	Object	Objeto de mensaje de ubicación
location.latitud	String	Latitud
location.longitud	String	Longitud
location.placeName	String	Nombre del lugar. Sera desplegado en el mensaje
location.address	String	Dirección. Sera desplegada en el mensaje

### 6.1.2. Informes de entrega

Un reporte de entrega contiene el estatus y el estado de cada mensaje enviado por la API de Broadcasterbot

```
{
  "statuses": [
    {
      "status": "success",
      "state": "delivered",
      "messageId": "fe93211f-9f28-426f-853d-6f85b3f53be6",
      "recipient": {
        "phone": "5544332211",
        "country": "MX"
      }
    }
  ]
}
```

Nombre	Tipo	Descripción
status	String	El estatus del mensaje “success” o “failure”
state	String	El estado del mensaje ( <a href="#">Ver tabla de estados</a> )
messageId	String	El id del mensaje al que pertenece el reporte
recipient	Object	Objeto del número al que pertenece el reporte
recipient.phone	String	Número local
recipient.country	String	Código del país del número del contacto (según ISO 3166 Alfa-2)

#### 6.1.2.1. Tabla de estados

Nombre	Descripción
queued	El mensaje ha sido recibido y encolado por la API de Broadcasterbot
dispatched	El mensaje ha sido enviado por la API de Broadcasterbot a los servidores de WhatsApp
sent	El mensaje ha sido enviado por WhatsApp al usuario final
delivered	El mensaje ha sido entregado exitosamente por WhatsApp al usuario final
read	El mensaje ha sido leído por el usuario final en la aplicación de WhatsApp
deleted	El mensaje ha sido eliminado o ha expirado en la aplicación
failed	El mensaje falló
no_opt_in	El mensaje ha sido rechazado porque el número receptor no ha sido dado de alta
no_capability	El mensaje ha sido rechazado por la API de Broadcasterbot ya que el número receptor no tiene capacidad de WhatsApp

## 6.2. Seguridad

El callback proporcionado a Concepto Móvil debe contar con un certificado de seguridad no autofirmado para confirmar la seguridad de la comunicación entre el cliente y la API de WhatsApp.

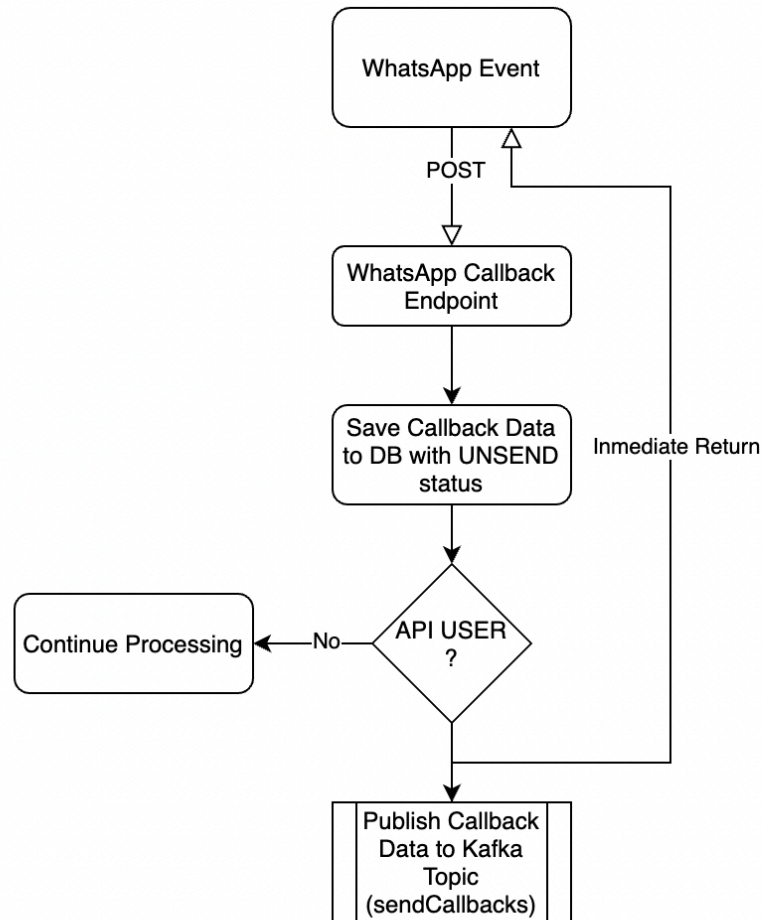
## 6.3. Manejo de colas

WhatsApp provee 2 tipos de eventos, eventos de Notificación y eventos de Status. Con la finalidad de resumir el funcionamiento del suministro de estos eventos, en adelante y en los diagramas se referirá a estos como “notificación”, “notification” o “callback data”

La plataforma utiliza la tecnología de Apache Kafka para el procesamiento asíncrono de estos eventos, siendo un Tópico de Kafka (Kafka Topic) una estructura de datos que funciona a manera de Cola (First In First Out).

### 6.3.1. Recepción de Notificaciones de WhatsApp

Cuando se genera una nueva notificación, WhatsApp consume una URL pública de BroadcasterBot (Callback Endpoint). Es en este momento cuando se guarda la información de dicha notificación en la Base de Datos y la información se escribe en el sistema de Colas (Kafka Topic), para un procesamiento asíncrono en segundo plano y se le notifica de regreso a WhatsApp que la información fue recibida satisfactoriamente.

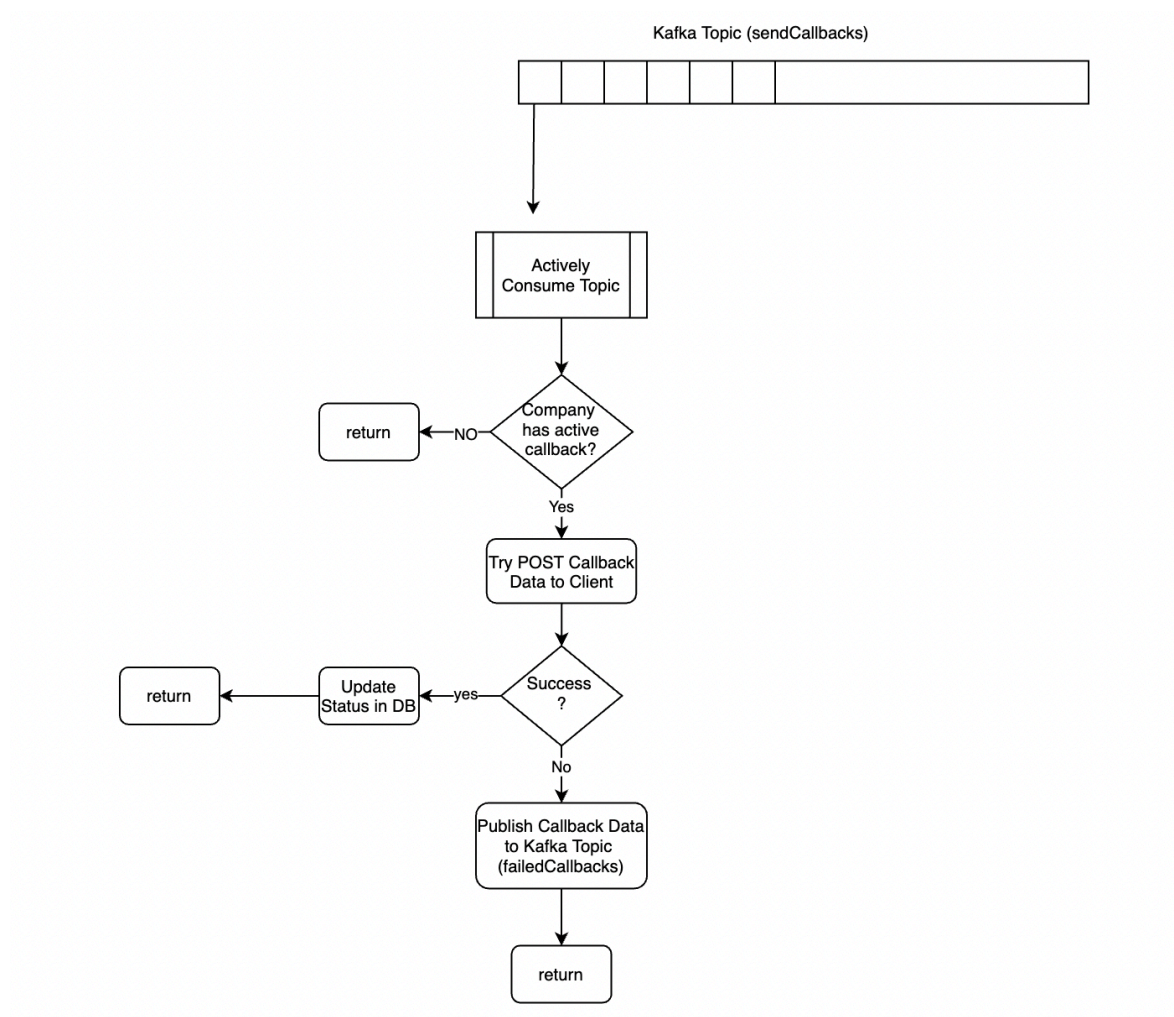


### 6.3.2. Procesamiento asíncrono de Notificaciones de WhatsApp

Una vez que la notificación fue escrita en el Tópico de Kafka, existe un consumidor activo que estará escuchando toda nueva notificación de este tópico.

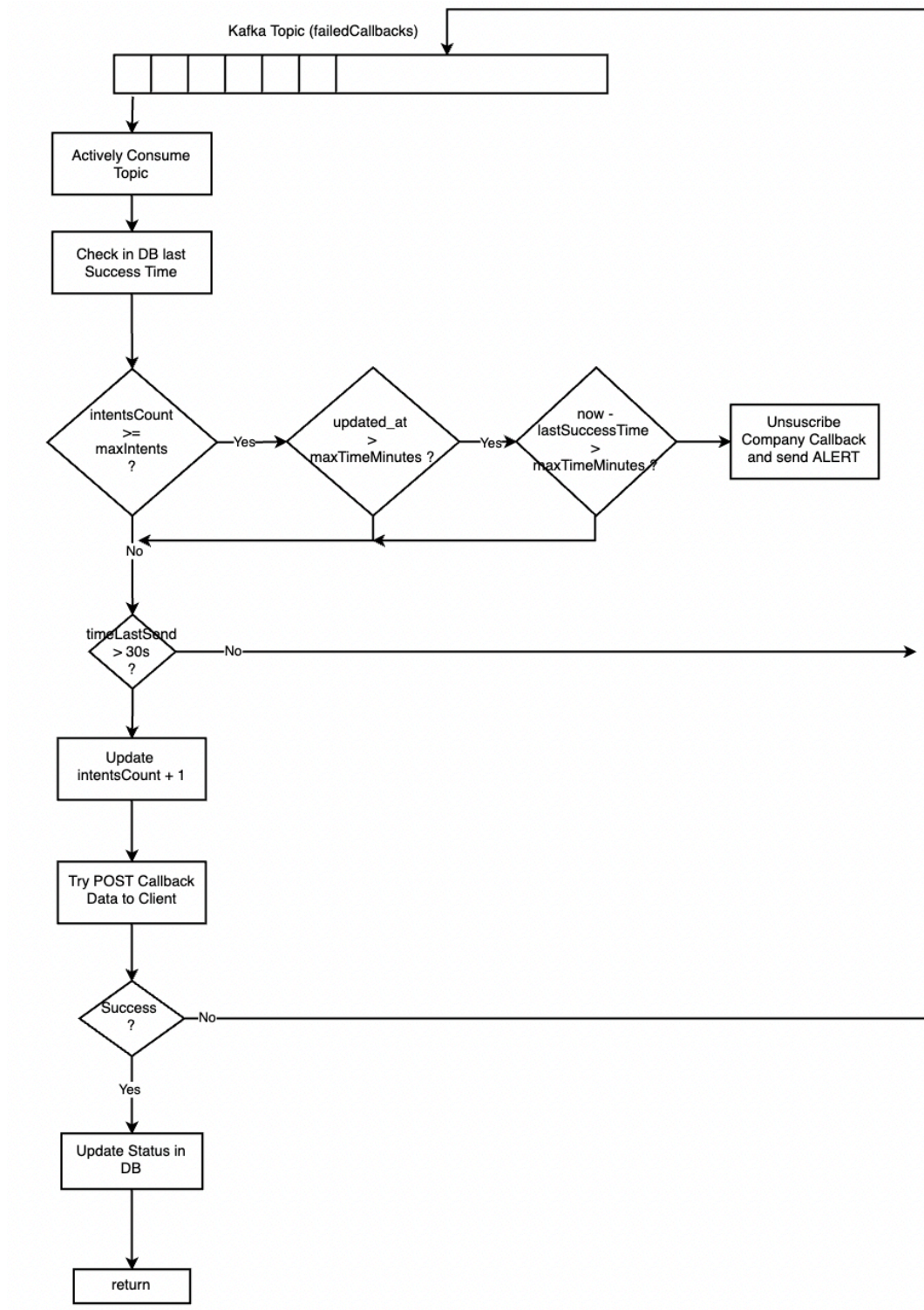
Este consumidor activo se encargará de realizar 3 acciones fundamentales enlistadas a continuación:

1. Verificar que la compañía tenga su Callback activo
2. Realizar el primer intento de envío de la notificación al Callback del cliente.
3. Actualizar el status de la notificación con el resultado de la operación anterior.



En el caso en que el resultado de la tercera operación sea exitoso, se actualiza el estatus de esta notificación y finaliza, de lo contrario cuando la respuesta del callback del cliente no sea exitosa, ésta detonará un segundo procesamiento de la notificación.

### 6.3.3. Procesamiento asíncrono de notificaciones fallidas





Cuando el callback del cliente no responde de manera exitosa en el primer intento de envío de la notificación, se detona un proceso secundario asíncrono, en el que se intenta de manera recurrente el reenvío de dicha notificación. Durante este proceso pueden ocurrir 1 de 3 acontecimientos:

1. Se envía satisfactoriamente la notificación
2. La notificación expira
3. Se desuscribe el callback del cliente y no recibirá nuevas notificaciones

Este proceso se rige por 5 parámetros descritos a continuación:

1. Max Intents (Número máximo de intentos, valor default: 5 minutos)
2. Max Time Minutes (Tiempo máximo en minutos, valor default: 5 minutos)
3. Expiration Time Minutes (Tiempo de expiración en minutos, valor default: 5 minutos)
4. Beat Time (Tiempo entre cada envío, valor default: 30 segundos)
5. Success Minutes Window (Ventana en minutos último exitoso, valor default: 5 minutos)

El parámetro "Success Minutes Window" describe la cantidad de tiempo en minutos, que se considera como "ventana de tiempo aceptable" durante la cual a pesar de existir notificaciones fallidas, el callback de la compañía no se desuscribir.

Esto puede ocurrir cuando el callback del cliente responde con un código de error a una notificación con un dato específico en el cuerpo de esta, sin embargo otras notificaciones las seguirá recibiendo de manera correcta. En este caso entra en juego el siguiente parámetro "Expiration Time Minutes" que describe el tiempo en minutos que la notificación estará "viva" o vigente para su envío recurrente. Una vez excedido este tiempo, la notificación expirará y no se enviará más.

Cuando todas las notificaciones fallan de manera consecutiva, entra en juego los parámetros "Max Intents" y "Max Time Minutes" que definirán la cantidad de intentos y el tiempo máximo que se intentarán enviar las notificaciones, que una vez excediendo estos ambos, provocará la desuscripción del callback de la compañía y con ello una alerta para el administrador de operaciones de la plataforma.

El parámetro "Beat Time" define el tiempo con el que se enviará una y otra vez la notificación fallida.