

API Hey Now!

Para Clientes



Contenido

Contenido	1
Introducción	2
Configuración de canales	2
Pruebas	2
Autenticación	3
Consulta de contactos de la agenda	4
Envío de mensajes	6
Consulta de estado de un envío	9
Contactos últimas 24hs	11
Datos Externos	12
Reporte de sesiones	14
Scroll	16
Uso de swagger ui	17
Autenticación	17
Anexo	19
Formato de reporte de sesiones	19

Introducción

El presente documento contiene información técnica sobre la interfaz de programación (API) de la plataforma Hey Now.

Dicha API está diseñada para que sistemas de terceros puedan interactuar con Hey Now con distintos fines, por ejemplo, enviar mensajes a través de los canales contratados.

A nivel técnico, se trata de una API REST y se utiliza el framework Swagger (<https://swagger.io>) como herramienta de diseño, documentación y testing, el cual implementa el standard OpenAPI Specification (OAS).

En la definición de los métodos se presentarán ejemplos (curl) de los casos típicos

Configuración de canales

Los diferentes canales que los clientes pueden contratar en Hey Now (por ejemplo números de WhatsApp) se configuran según los acuerdos comerciales de cada caso. Algunas de esas configuraciones incluyen la posibilidad de definir si un determinado canal debe validar que un contacto pertenece a su agenda, es decir que ese contacto le ha escrito previamente; otra configuración posible es definir un canal alternativo asociado al primero. Esto último típicamente se utiliza cuando se define un canal entrante y otro saliente.

Pruebas

El punto de entrada a la API es: <https://web.heynew.com.uy:8310/api>, y puede encontrarse la especificación completa en el documento JSON que se encuentra en <https://web.heynew.com.uy:8310/swagger.json>, con el cual se pueden utilizar herramientas de generación de código automático como por ejemplo la que dispone Swagger (<https://swagger.io/swagger-codegen>).

Para realizar pruebas, se dispone del sitio <https://web.heynew.com.uy:8310> en donde pueden utilizarse los diferentes métodos que provee el Api

Autenticación

Todas las solicitudes al api, deben de tener el token de autenticación ([bearer token](#)). Para obtener dicho token se debe de ejecutar el método de login del api (único método para el cual no es requerida la autenticación). La validez de un token está dada, luego de su solicitud hasta un máximo de 24 horas.

method	POST	
path	/api/login	
parametros	El cuerpo de la solicitud contiene un json con los siguientes parámetros	
	name	login del usuario
	password	contraseña del usuario a registrarse

Códigos de error

401	No autorizado (usuario o contraseña incorrecta)
500	Error interno del servidor

Solicitud:

```
curl -X POST "https://web.heynew.com.uy:8310/api/login"
-H "accept: application/json"
-H "Content-Type: application/json"
-d "{\"name\":\"login\",\"password\":\"myPassword\"}"
```

Retorno (body):

```
{
  "success": true,
  "message": "logged",
  "token": "...Token..."
}
```

Consulta de contactos de la agenda

En caso de existir un contacto con un identificador particular, retorna información sobre el mismo, en caso contrario retorna el estado http correspondiente¹.

method	GET	
path	/api/contact/	
parametros	En la query se deben de proporcionar los datos del contacto	
	channel	identificador del canal
	platformId	identificador de la plataforma
	clientId	Identificador del cliente. Este parámetro es opcional, si la consulta se realizar sin un cliente particular se retorna toda la agenda, lo que puede ocasionar demoras muy grandes en la consulta.

Códigos de error

401	Token no válido o autenticación no realizada
403	No se pudo acceder al platformId solicitado o el mismo no existe
404	Contacto no encontrado
500	Error interno del servidor

¹ https://es.wikipedia.org/wiki/Anexo:C%C3%B3digos_de_estado_HTTP

Solicitud:

```
curl -X GET "https://web.heyknow.com.uy:8310/api/contact/channel/platformId/clientId"  
-H "accept: application/json"  
-H "Authorization: Bearer ...Token..."
```

Retorno (body):

```
{  
  "id": "5ae08c73b8815f4a034dec24",  
  "phones": [ "2222222222" ],  
  "first_name": "2222222222",  
  "last_name": null,  
  "profile_pic": null,  
  "locale": null,  
  "timezone": null,  
  "gender": null,  
  "channels": [ {  
    "channel": 1,  
    "platformId": "2222222222",  
    "clientId": "2222222222",  
    "metaData": {}  
  }  
]  
}
```

Envío de mensajes

Se envía el mensaje al cliente dado, hey creará un identificador de la solicitud de envío idMessageHey, con el cual se puede consultar el estado del envío.

method	POST		
path	/api/message/		
parametros url	En la query se deben de proporcionar los datos del contacto		
	channel	identificador del canal	
	platformId	identificador de la plataforma	
	clientId	Identificador del cliente.	
parametros body	En la query se deben de proporcionar los datos del contacto		
	text	Texto a enviar (parámetro excluyente con el parámetro file)	
	idMessageHey	Identificador del mensaje, es recomendable solo utilizar este parámetro si es indispensable, de lo contrario la plataforma retorna uno para dar seguimiento a la solicitud	
	file	Archivo a enviar (parámetro excluyente con text)	
		data	Contenido del archivo (generalmente en base 64)
		mimeType	Mime type del archivo
		encode	Codificación del archivo (por el momento solo se soporta base64)
		name	Nombre del archivo (usualmente es el nombre propuesto al momento de la descarga)
	HSM	Archivo a enviar (parámetro excluyente con text)	
		template	template del hsm
		params	Arreglo de strings con los parámetros utilizados, en orden de aparición, comenzando por el 1 {{1}}
		languagePolicy	parametro opcional, generalmente se utiliza “DETERMINISTIC”
		languageCode	Lenguaje en el que se envía el mensaje, este parámetro como el

			anterior son determinados al momento de solicitar el HSM.
--	--	--	---

Códigos de error

401	Token no válido o autenticación no realizada
403	No se pudo acceder al platformId solicitado o el mismo no existe
500	Error interno del servidor

Solicitud:

```
curl -X GET "https://web.heyknow.com.uy:8310/api/contact/channel/platformId/clientId"  
-H "accept: application/json"  
-H "Authorization: Bearer ...Token..."
```

Retorno (body):

```
{  
  "_id": "5ae08c73b8815f4a034dec24",  
  "phones": [ "22222222222" ],  
  "first_name": "22222222222",  
  "last_name": null,  
  "profile_pic": null,  
  "locale": null,  
  "timezone": null,  
  "gender": null,  
  "channels": [ {  
    "channel": 1,  
    "platformId": "22222222222",  
    "clientId": "22222222222",  
    "metaData": {}  
  }  
]  
}
```

Consulta de estado de un envío

Se retorna el estado de un mensaje, identificándolo por idMessageHey.

method	GET	
path	/api/message/	
parametros url	En la query se deben de proporcionar los datos del contacto	
	channel	identificador del canal
	platformId	identificador de la plataforma
	idMessageHey	Identificador del envío.
parametros query string		
	verbose	Retorna información extra sobre el estado del mensaje (opcional)

Se retorna el código de estado de envío (string), el mismo puede ser uno de los siguientes:

Código	Descripción
F	El envío del mensaje fallo, esto puede ser porque el contacto no pasó la validación de estar en la agenda o existe.
?	El mensaje se encuentra encolado para su envío, pero todavía no se realizó.
RS	El mensaje se envió al canal correspondiente.
RU	Se confirmó la recepción del mensaje por parte del usuario.
LU	Se confirmó la lectura (apertura) del mensaje por parte del usuario.

En caso de enviar el parámetro verbose en true se retorna un json con el estatus y la descripción en caso de error:

```
{
  "status": "F",
  "message": "Error en el envío ....",
}
```

Códigos

401	No autorizado (usuario o contraseña incorrecta)
403	No se pudo acceder al platformId solicitado o el mismo no existe
500	Error interno del servidor

Solicitud:

```
curl -X GET "https://web.heynew.com.uy:8310/api/message/channel/platformId/idMessageHey"
-H "accept: application/json"
-H "Authorization: Bearer ...Token..."
```

Retorno (body):

```
"F"
```

Contactos últimas 24hs

Se retorna la lista de contactos que le escribieron al bot en las últimas 24 hs.

method	GET					
path	/api/contacts24Hs/{channelId}/{platformId}					
parametros url	En la query se deben de proporcionar los datos del bot al que los contactos le escribieron. <table><tr><td>channelId</td><td>identificador del canal</td></tr><tr><td>platformId</td><td>identificador de la plataforma</td></tr></table>		channelId	identificador del canal	platformId	identificador de la plataforma
channelId	identificador del canal					
platformId	identificador de la plataforma					

Solicitud:

```
curl -X GET "https://web.heynew.com.uy:8310/api/contacts24Hs/channel/platformId"
-H "accept: application/json"
-H "Authorization: Bearer ...Token..."
```

Retorno (body):

```
[
  {
    "_id": "5e430e753787517cdc841597",
    "contact": {
      "channels": [
        {
          "metaData": {},
          "clientId": "59891007188",
          "channel": 1,
          "platformId": "59898507777"
        }
      ],
      "gender": null,
      "timezone": null,
      "profile_pic": null,
      "last_name": null,
    }
  }
]
```

```

    "phones": [
      "59891007188"
    ],
    "_id": "5e430cd93787517cdc83fd86",
    "creationDate": "2020-02-11T20:21:45.551Z",
    "locale": null,
    "first_name": "59891007188"
  }
}
]

```

Códigos

401	No autorizado (usuario o contraseña incorrecta)
403	No se pudo acceder al platformId solicitado o el mismo no existe
500	Error interno del servidor

Solicitud:

```

curl -X GET "https://web.heynow.com.uy:8310/api/message/channel/platformId/idMessageHey"
-H "accept: application/json"
-H "Authorization: Bearer ...Token..."

```

Retorno (body):

```
"F"
```

Datos Externos

Se envían una actualización de los datos en tiempo real de una sesión y se dispara una acción de externalDataUpdate para que el bot reaccione acorde.

method	POST								
path	/session/data/:channel/:platformId/:contactId/:sessionId								
parametros url	<p>En la query se deben de proporcionar los datos del contacto</p> <table> <tr> <td>channel</td><td>identificador del canal</td></tr> <tr> <td>platformId</td><td>identificador de la plataforma</td></tr> <tr> <td>contactId</td><td>Identificador del cliente.</td></tr> <tr> <td>sessionId</td><td>Identificador de la sesión</td></tr> </table>	channel	identificador del canal	platformId	identificador de la plataforma	contactId	Identificador del cliente.	sessionId	Identificador de la sesión
channel	identificador del canal								
platformId	identificador de la plataforma								
contactId	Identificador del cliente.								
sessionId	Identificador de la sesión								

parametros body	Los parametros del objeto que se envíe en el body se cargarán en la propiedad "session.data" de la sesión. Si ya existe un parametro con el mismo nombre, se sobrescribirá.
----------------------------	---

Códigos de error

403	No se pudo acceder al platformId solicitado o el mismo no existe
500	Error interno del servidor

Solicitud:

```
curl -X POST
"https://web.heynow.com.uy:8310/session/data/1/59891668668/59892993471/
2020020920521083758"
-H "accept: application/json"
-H "Content-Type: application/json"
-d "{\"prueba\":\"test\"}"
```

Retorno (body):

```
{
  "status": "OK",
}
```

Reporte de sesiones

Se retornan las sesiones en un determinado lapso de tiempo. La respuesta se retorna en páginas, y para avanzar entre ellas se utilizan scrolls.

Por ejemplo, si la consulta retorna 1000 registros en páginas de 50, se tomarán los primero 50 registros del parámetro data y luego con el parámetro scroll se realizará una consulta al método “scroll” con el identificador dado hasta que no hayan más datos retornados.

method	GET	
path	/api/report/session	
parametrosquery string	begin	Fecha de finalización de sesión mínima para ser retornada en la consulta (opcional) En formato ISO 8601
	end	Fecha de finalización de sesión máxima para ser retornada en la consulta (opcional) En formato ISO 8601
	pageSize	Tamaño máximo permitido para cada consulta, el valor debe de estar entre 1 y 500. Parámetro opcional, valor por defecto 50.

Ejemplo de retorno:

```
{
  "scroll": {
    "id": "xxxx",
    "length": 10000,
    "total": 3
  },
  "data": [{...},{...},{...}]
}
```

Se retorna un elemento scroll que contiene:

id: identificador del scroll para continuar la consulta

length: cantidad de datos retornados en la consulta (coincide con el tamaño size cuando hay más elementos en el scroll)

total: total de mensajes que retorna la consulta luego de completar todos los scroll.

El scroll tiene una validez de 1 minuto luego de cada consulta.

El elemento **data** contiene la información consultada

Códigos

200	OK
401	Token no válido o autenticación no realizada
403	Permisos insuficientes
500	Error interno del servidor

Scroll

Se retorna la continuación a una consulta realizada, con un identificador de scroll dado.

method	GET			
path	/api/report/{scrollId}			
parametros url	<table><tr><td>scrollId</td><td>identificador del scroll</td></tr></table>		scrollId	identificador del scroll
scrollId	identificador del scroll			

Ejemplo de retorno:

```
{
  "scroll": {
    "id": "xxxx",
    "length": 10000,
    "total": 3
  },
  "data": [{...},{...},{...}]
}
```

Se retorna un elemento scroll que contiene:

id: identificador del scroll para continuar la consulta

length: cantidad de datos retornados en la consulta (coincide con el tamaño size cuando hay más elementos en el scroll)

total: total de mensajes que retorna la consulta luego de completar todos los scroll.

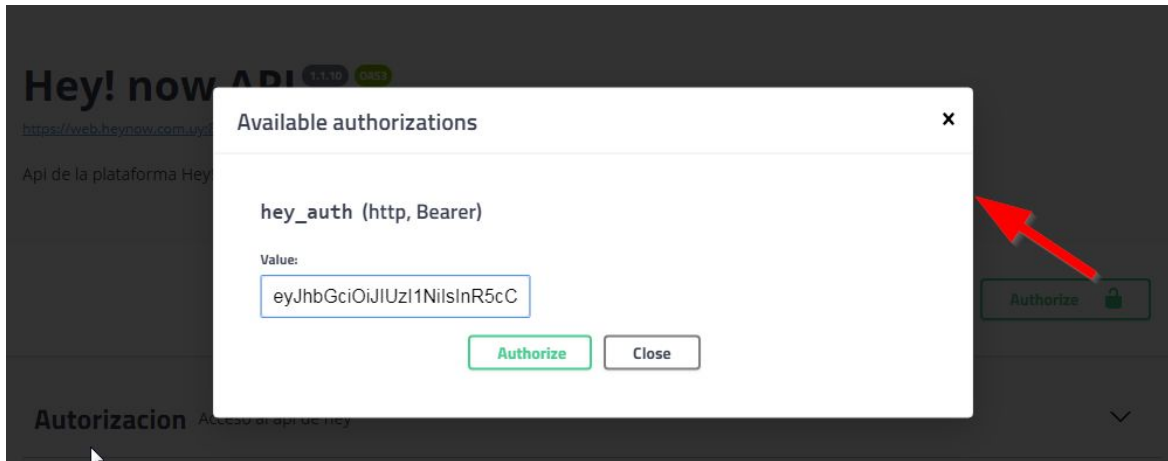
El scroll tiene una validez de 1 minuto luego de cada consulta.

El elemento **data** contiene la información consultada

Códigos

401	No autorizado (usuario o contraseña incorrecta)
403	No se pudo acceder al platformId solicitado o el mismo no existe
404	El scroll no fue encontrado o vencido
500	Error interno del servidor

Se debe de copiar el token y pegarlo en el diálogo que se despliega al clickear en el botón “Authorize” que se encuentra en la parte superior derecha del sitio.



Luego de realizar esta autorización se pueden ejecutar todos los métodos expuestos en la página (siguiendo las indicaciones en la misma).

Es importante notar que el token tiene una validez de 24Hs y pasado este lapso se deberá ejecutar el login nuevamente.

Anexo

Formato de reporte de sesiones

A continuación se detallan los datos más relevantes que se retornan en el método de reportes de sesión. Es importante tener en cuenta que este reporte retorna información de la sesión tal cual la administra el sistema de hey, por lo que hay datos que en un primer instancia pueden no tener utilidad para los objetivos de quien consulta, por temas de performance estos datos

Los datos son retornados como JSON, donde podemos encontrar dos propiedades

scroll

Retorna información para continuar la consulta, si el largo de los datos retornados excede el tamaño de la página. para continuar la consulta se debe de utilizar el parámetro **id**. También se retorna información sobre el tamaño total de la consulta (si se realizan todos los scroll) así como el largo de los datos retornados.

```
{
  "scroll": {
    "id": "DnF1ZXJ5VGhlbkZ1dGN0BQAAAAAafgLFkxpd1VESzN1UjZ1UG82Zmx",
    "length": 50,
    "total": 706760
  },
}
```

data

El segundo parámetro que se retorna en la consulta es **data**, éste es un arreglo que contiene las sesiones que corresponden a la consulta realizada.

Cada elemento de data contiene:

_id:

Identificador del **registro** de sesión, este identificador es único en el sistema y permite identificar las sesiones.

_source:

Dentro de source se encuentran los datos correspondientes a la sesión.

```
"data": [{
  "_index": "hey.log_session",
  "_type": "_doc",
  "_id": "5d94ff13[REDACTED]5429e44",
  "_score": 16.053394,
  "_source": {
    "beginSession": "2019-10-02T19:38:20.941Z",
    "channel": 4,
    "chat": [{
      "clientId": "528b2d50[REDACTED]ad",
      "contact": {
        "endSession": "2019-10-02T19:48:35.857Z",
        "messages": {
          "platformId": "[REDACTED]",
          "queryData": {
            "session": "16d8d[REDACTED]f7eb",
            "sessionLength": 614916,
            "steps": [{
              "versions": {
                "chat": 1,
                "contact": 1,
                "messages": 1,
                "steps": 1
              }
            ]
          }
        }
      }
    ]
  }
}, {
```

data \ _source

En este nodo se encuentra la información propia de una sesión de hey, se puede encontrar tanto información cruda como “agregada”, de manera que hay cálculos y/o conteos realizados.

Se analizará la información más relevante de las sesiones, algunos parámetros dependen del desarrollo que se realice para el cliente particular y/o del bot.

channel:

Identificador del canal (1 o 6 whatsapp, 2 Messenger, 3 twitter, 4 webchat, 5 facebook, 7 Instagram, 8 Mercado Libre).

platformId:

Identificador de la plataforma, esto es un identificador para el canal de comunicación, por ejemplo, para whatsapp (channel 1 0 6) es el número de teléfono entrante. La clave channel y platformId también es lo que en hey llamamos “Bot”, por más que haya atención humana en exclusividad, si el número es administrado por hey siempre hay un bot asociado.

clientId:

Identificador del cliente para el bot (channel, platformId), por ejemplo, para whatsapp el valor coincide con el número de teléfono del contacto que se comunicó con el bot. En otros canales, por ejemplo Facebook, este valor difiere entre los diferentes platformId, no así con Whatsapp.

session:

Es el identificador de la sesión (no el identificador del registro), este identificador no es necesariamente único en el sistema, ya que existen canales que pueden definir este valor. Si es único dentro de la clave compuesta (channel, platformId, clientId, session).

companyId:

Este es el identificador de la cuenta creada en hey para administrar los diferentes bots, cada login corresponde a una y solo una compañía.

beginSession:

Es el momento en el que comenzó la sesión, o sea cuando llegó el primer mensaje en el sistema, por ejemplo que el contacto envió hola al número de visanet.

endSession:

Es el momento en que finaliza la sesión, o sea cuando la sesión es eliminada.

startPanelDate:

Es el momento en que la conversación pasó a panel (o sea que espera a ser atendida).

firstAgentContact:

Es el momento en la que la conversación tuvo actividad de un agente.

contactLastInteractionDate:

Es el momento en que el contacto envió su último mensaje, dentro de esta interacción, con la excepción que si el contacto nunca se comunicó es la fecha de su última interacción previo a la sesión. Esto es un caso excepcional, ya que sería una sesión creada de manera proactiva y en la que el contacto nunca participó.

firstIncomingMessage:

Es el momento en que el contacto envió su primer mensaje dentro de la sesión.

sessionLength:

Duración de la sesión en milisegundos.

chat:

Son los mensajes/eventos que se enviaron y recibieron durante la sesión.

steps:

Son los pasos (marcas) que se realizaron durante la sesión.

messages:

Es Información sobre los mensajes cursados que se calculo al momento de finalizar la sesión, con el fin de simplificar los cálculos.

messages\incoming:

Contiene cuantos mensajes entrantes hubieron y si fueron en modo "panel" o en modo "bot", para el caso panel además se detalla la habilidad.

messages\outcoming:

Al igual que el punto anterior controla cuántos mensajes salientes se enviaron, discriminando por el modo ("bot" o "panel") y la habilidad, en el caso de ser un mensaje del panel además se registra información del agente que se comunicó con el.

messages\incomingMessages:

Total de mensajes entrantes.

messages\outcomingMessages:

Total de mensajes salientes

messages\totalMessages:

Total de mensajes

contact:

Información del contacto al momento de finalizar la sesión, esto es, una “foto” de la información del contacto cuando se comunicó con el bot. No es necesariamente la información más actual del contacto. Esto último se puede consultar con el método de consulta de información de contactos.

versions:

Es la versión del formato de la sesión, esto es, como se presentan los formatos y la información que se encuentra “calculada”.

queryData:

Información de las tipificaciones, la información aquí presentada dependerá del bot.