

# Genesys Universal Messaging User Guide

Gildas Cherruel (gildas.cherruel@genesys.com)

v. 1.0.20-4-gea11f25, 2023/10/24

# Contents

<b>1 Concepts</b>	<b>3</b>
<b>2 Social Media Configuration</b>	<b>5</b>
Apple Messages for Business	5
Disqus	6
Google Business Messages	7
Google Chat for Workspace	8
LINE	10
Using LINE's Call To LINE	13
Microsoft Teams	15
Slack	17
Telegram	19
Viber	20
Zalo	21
<b>3 Architect Inbound Flows</b>	<b>25</b>
<b>4 Canned Responses</b>	<b>27</b>
Create a Canned Response	27
Send a Canned Response	28
Complex Canned Responses	29
Templated Canned Responses	31
Samples	33
Genesys	33
Apple Messages for Business	33
Google Chat for Workspaces	37
Google Business Messages	40
LINE	42
Microsoft Teams	45
Slack	47
Viber	50
Zalo	51



# Chapter 1

## Concepts

Genesys Universal Messaging is micro-service based application that offers a platform to ease message communication between Genesys Cloud and various social media providers.

Each micro-service is responsible for exchanging messages between one social media provider and the Universal Messaging platform.

Each micro-service is multi-tenant and manages connections to many accounts for the social media it takes care of.

Universal Messaging is deployed on a Kubernetes cluster allowing efficient management of all the micro-services as well as seamless scalability.

Here is the list of the social media providers supported at present:

- Apple Messages for Business Connector, interfaces with [Apple Messages for Business](#)
- Bedore Connector, interfaces with the Bedore Bot;
- BizM for KakaoTalk, interfaces with the [KakaoTalk](#) Social Media via [BizM](#);
- Disqus Connector, interfaces with [Disqus](#) Comment Service;
- Google Business Messages Connector, interfaces with [Google Business Messages](#);
- Google Chat Connector, interfaces with [Google Chat](#);
- Infobank for KakaoTalk, interfaces with the [KakaoTalk](#) Social Media via [Infobank](#);
- LINE Connector, interfaces with the [LINE](#) Social Media;
- Media4U Connector, interfaces with the [Media4U](#) SMS Services;
- Microsoft Teams Connector, interfaces with [Microsoft Teams](#);
- Slack Connector, interfaces with the [Slack](#) Social Media;
- PlusMessage Connector, interfaces with [DOCOMO +Message](#) Social Media;
- Telegram Connector, interfaces with the [Telegram](#) Social Media;
- Viber Connector, interfaces with the [Viber](#) Social Media;
- WeChat Connector, interfaces with the [WeChat](#) Social Media;
- Zalo Connector, interfaces with the [Zalo](#) Social Media;

None of micro-services store any end-user sensitive information and some of them will store attachments temporarily so the Genesys Cloud agent or the end-user can view them. Once the conversation is done, these temporary files are deleted.

Universal Messaging is deployed and maintained by Genesys on our Amazon Web Services accounts.

Our customers have nothing to install at their premises nor in their clouds. They are, however, responsible for creating/buying accounts with the social media providers they want to use with Universal Messaging.

With the social media providers, end-users will use their provider client the same way they are used to.

With Genesys Cloud, agents will use the Messaging UI the same way they are used to. Please refer to the official documentation from Genesys about the configuration of these agents, the queues, Architect flows, Canned Responses.



## Chapter 2

# Social Media Configuration

In the next paragraphs, we will explain how to configure some of the social media providers.

### Apple Messages for Business

Customers should go to [Apple Register](#). They will need an Apple ID, if they do not have any, they can get one from the [Create Your Apple ID](#) page. once that is done, they should connect, choose “Messages for Business Accounts”, and create a Business Account:

The screenshot shows the 'Genesys' header at the top. Below it is the 'Your Connections' section, which includes a 'Manage' link. There are three connection cards: 'Messages for Business Accounts' (Messages), 'Indoor Maps' (Maps), and 'Public Wi-Fi Networks' (Settings). Below this is the 'Your Organization' section, which includes links for 'Locations' (Review your locations in Apple services), 'Members' (Add team members to assist with managing this organization), and 'Brands' (Review your publicly recognizable brand / trade names and logos).

Once the Business Account has been approved by Apple, the customer will need to link their Business Identifier to Universal Messaging. When configuring the Messaging Platform, they will enter the Universal Messaging register webhook URL, given by a Genesys Representative, <https://gum-xxxx.genesyscsp.com/amb/register>:

## Messaging Platform

Select the messaging platform to use for this Messages for Business account.

Apple authorized commercial messaging platform

Advanced ^

URL provided by your messaging platform

Select this option if your messaging platform provider gave a specific URL to connect your Messages for Business account.

Proprietary or internal platform

Select this option if your organization will self-manage message routing and agent desktop.

Cancel

Submit

They should be directed automatically to the Universal Messaging registration where they can enter their information so a Genesys Representative can finalize their configuration in Universal Messaging.

To configure Apple Pay, the customer will need to go to [Apple Pay](#), sign up with a Payment Service Provider and create a Merchant ID. They will need to provide their Merchant ID, name, certificate, Private Key, and Payment Gateway to Universal Messaging. The domain to use with the Payment Service Provider is the one used by Universal Messaging (the webhook hostname).

**Note:** Apple Pay configuration is still a bit experimental and will need manual intervention from the deployment team. This is due to the fact no real payment gateway is available for testing.

To configure the Integrated OAuth2 Authentication, the customer will need to sign up with an OAuth2 provider (like [Google](#) or [Auth0](#)). They will need to provide their response type, OAuth scope, and Client Secret to Universal Messaging.

## Disqus

Customers should go to [Disqus.com](#)'s website and create an account. The customers should send to their Genesys representative the API Key, the API Secret, and the Access Token.

To send comments to Universal Messaging, the customer should add some Javascript code like this one:

```
var disqus_config = function () {
  this.page.url = 'https://PAGE_URL/disqus.html'; // Replace PAGE_URL with your page's canonical URL
  this.page.identifier = ''; // Replace PAGE_IDENTIFIER with your page's unique identifier variable
  // See: https://help.disqus.com/en/articles/1717088-capture-disqus-commenting-activity-via-callbacks

  this.callbacks.onNewComment = [
    async (comment) => {
      try {
        console.log('Guest posted a comment: ', comment);
        // ***** Replace YOUR-FQDN with the your Universal Messaging canonical URL
        let response = await fetch('https://gum-xxxx.genesyscsp.com/disqus', {
          method: 'POST',
          redirect: 'follow',
          body: JSON.stringify({
            tenantId: '9606502f-dead-beef-feed-1b3a33129ed2', // *** Add the Identifier from Genesys
            postId: comment.id
          })
        });
        console.log('Response: ', response);
      }
    }
  ];
};
```

```
    } catch (err) {  
      console.error('Failed to send comment to Universal Messaging', err);  
    }  
  }  
}  
];  
};
```

The exact URL for Universal Messaging will be given by a Genesys representative.

## Google Business Messages

Customers will need a Google Account on [Google Cloud](#) that has some Billing attached.

Once they have that, they should go to the [Google Business Messages Console](#) and click on the **Create partner account** button.

Fill in the form:

**Create a Business Messages partner account**

A Business Messages partner account allows you to create Business Messages brands and agents. [Learn more](#)

Looking for [RCS Business Messaging](#), [Verified Calls](#), or [Verified SMS](#)?

Corporate email

[Switch account](#) ?

Your name \*

Partner name \*  ?

Partner website \*  ?

Region \*

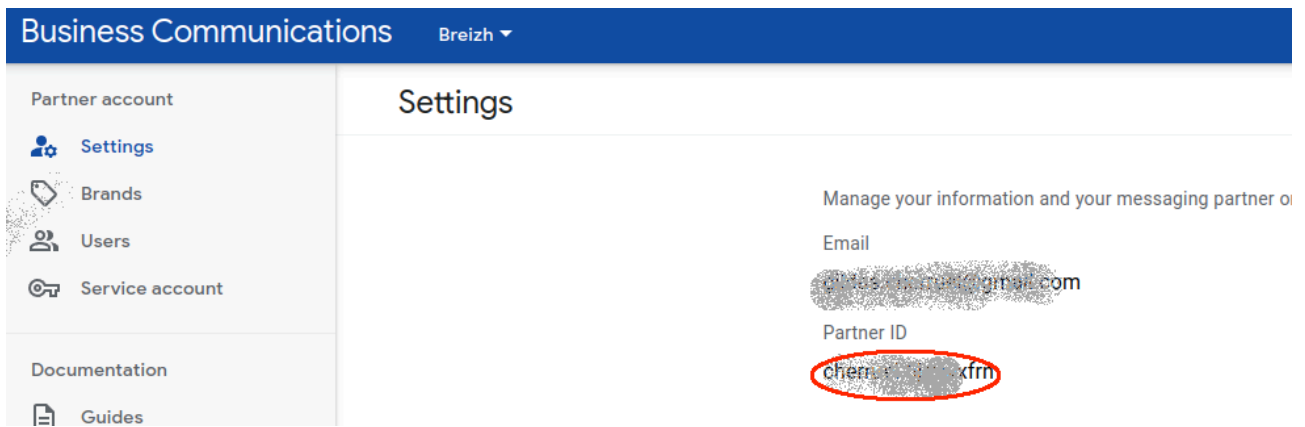
By clicking **Create** you agree to the [Terms of Service](#).

**Create**

Once the account is created, they should go to the **Partner account settings** page, and complete the Settings page. There is no need to set a webhook now. This will be done later when we add Google Agents.

They should send Genesys the **Partner Id** that comes from the Partner Settings page of the Google Console:





On the **Brands** page, they should create a new brand.

Then on the **Service account** page, they should create a key. The JSON file that is automatically downloaded should be kept safe and sent to the Genesys representative. Once Genesys has prepared the customer's configuration on Universal Messaging, they will contact the customer.

The next step is to create an **agent** and click on it to go to its configuration page. The **Agent information** should be configured to suit the customer's needs.

Finally, on to the **Integrations** page, and the customer should configure the **webhook** with the URL the Genesys representative gave them, then generate a **Client Token** and **verify** the webhook.

Testing can be done from the Agent Overview page of the [Google Business Messages Console](#) and click on the **Android**, **iOS** or **Send** button to get the instructions on how to start a conversation.

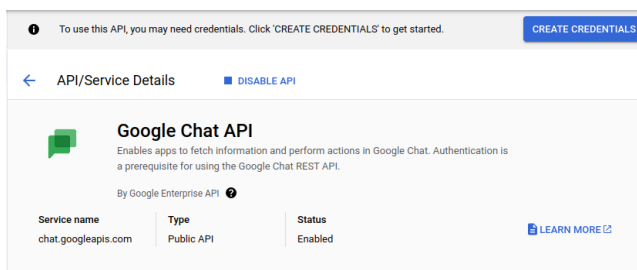
Once the testing is done, the agent should be **Verified** and **Launched** so real-life guests can connect.

## Google Chat for Workspace

To configure Google Chat, the customer needs to sign up for a [Google Workspace](#) account with access to [Google Chat](#). In the Google Cloud Project that governs your Google Workspace, they need to enable the [Chat API](#) and the [Drive API](#), if the users will be allowed to send documents to Universal Messaging via their Google Drive.

**Note:** The **"Billing"** must be set correctly for your Google Cloud Project.

Once enabled, the customer needs to create some credentials for Universal Messaging to use. On the [Google Cloud Console](#), after the Google Chat API is enabled, they should see a message asking you to create some credentials by clicking on the **Create Credentials** button and selecting **OAuth client ID**.



The customer should select **Application Data** as the data that will be accessed and **No, I'm not using them** for using GCE, GKE, or GCF:

## 1 Credential Type


### Which API are you using?

Different APIs use different auth platforms and some credentials can be restricted to only call certain APIs.

Select an API \*

### What data will you be accessing? \*

Different credentials are required to authorize access depending on the type of data that you request. [Learn more](#)

- User data   
 Data belonging to a Google user, like their email address or age. User consent required. This will create an OAuth client.
- Application data  
 Data belonging to your own application, such as your app's Cloud Firestore backend. This will create a service account.

### Are you planning to use this API with Compute Engine, Kubernetes Engine, App Engine, or Cloud Functions?

Applications running on GCE, GKE, GAE, and GCF can use Application Default Credentials and don't require that you create a credential.

- Yes, I'm using one or more
- No, I'm not using them

NEXT

Then, **Next** and give a name and an ID to the service account (and, eventually, a description):

## 1 Service account details

Service account name

Display name for this service account

Service account ID \*  
  

Email address: gum-gchat-demo@gum-gchat-demo.iam.gserviceaccount.com 

Service account description

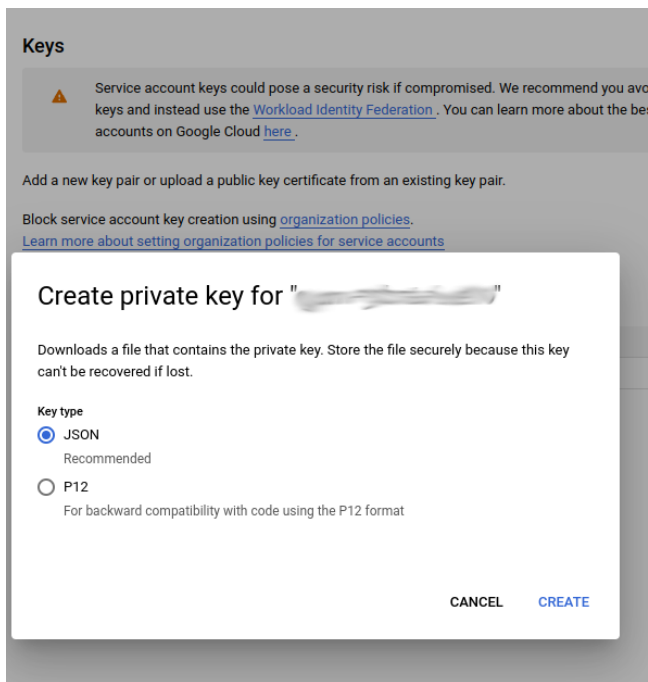
Describe what this service account will do

CREATE AND CONTINUE

Finally, they should click **Create and Continue** and **Done**. (We do not need more roles nor user access than the default ones.)

Back on the Google Chat API screen, in the **Credentials** section, they should see the new service account. There is a warning asking to configure the consent screen. They should click on the **Configure Consent Screen** button and choose **Internal** as the user type, configure the App information to their liking and click **Save**.

The next step involves clicking on the **Service Accounts** section and clicking on the service account that was just created. they should click on the **Keys** tab and click on the **Add Key** button, select **JSON** as the key type and click **Create**. A JSON file will be downloaded. It should be kept it safe and sent to their Genesys representative!



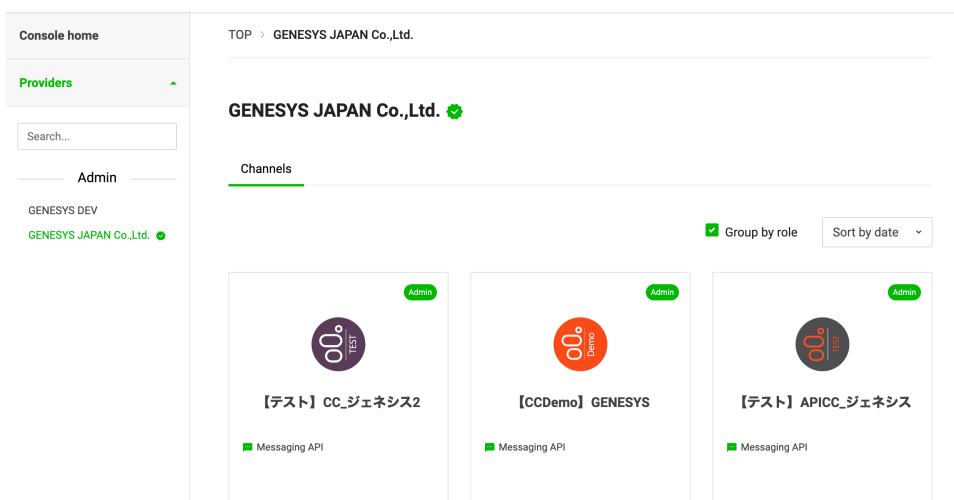
Finally, they should go back to the **Enabled APIs & Services** section, select the **Google Chat API**, and click on the **Configuration** tab of the Google Chat API, write down the App ID (the text is gray just under **Application Info**), give a name, an Avatar URL, and a description to the bot; add the functionalities for **Receive 1:1 messages** and **Join spaces and group conversations**; configure the **App URL** to the URL of your Universal Messaging deployment (something like `https://gum-xxxx.genesyscsp.com/gchat`, where `xxxx` is the actual Universal Messaging deployment); choose the **Permissions**, either everyone in the workspace organization or specific people; finally, click **Save**.

The customer should also send the App ID they received in the previous paragraph to their Genesys representative.

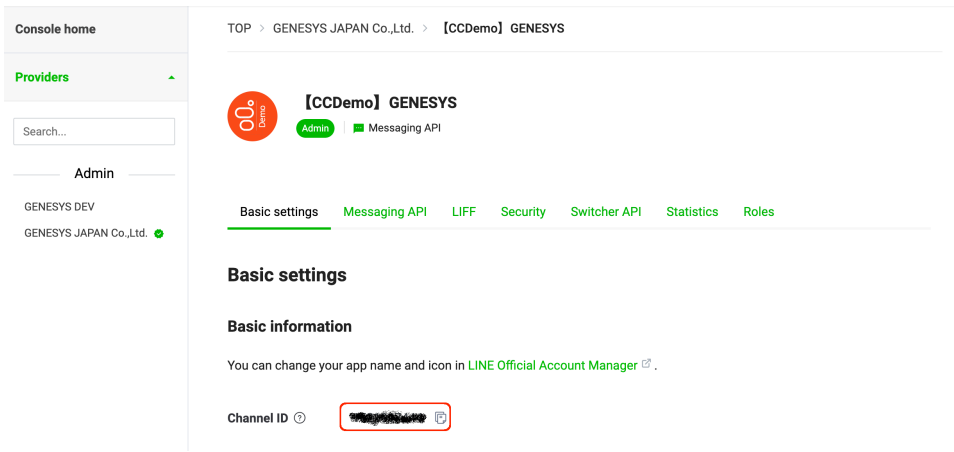
## LINE

To configure LINE with Universal Messaging, you will need a Business Account with **LINE Business**, you can create one at [LINE for Business](#).

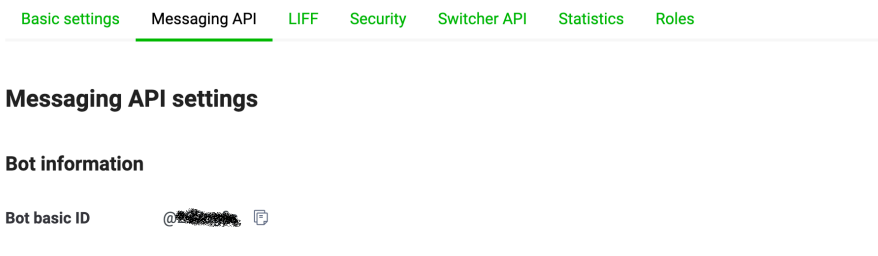
You should find on you LINE Business Account page all the channels that you have created, for example:



And pick the channel you will use with Universal Messaging. Note down the **Channel ID** and the **Channel Secret** (further down on the same web page) from the **Basic Settings** page:



Then, the **Bot basic ID** (with its @ sign) from the **Messaging API** page:

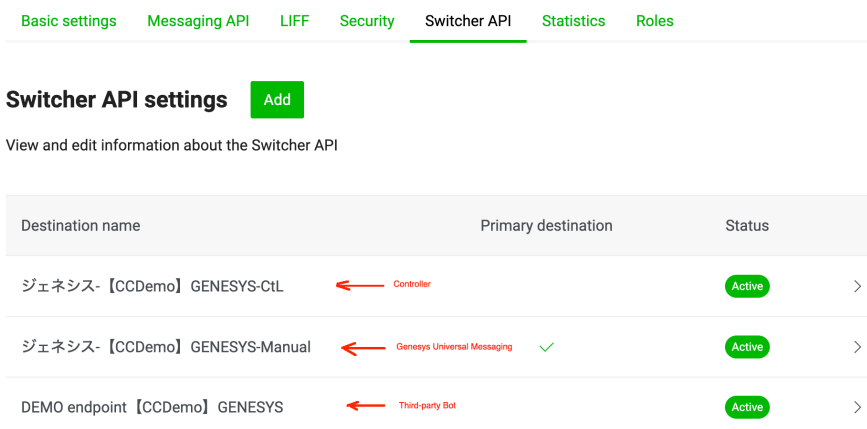


You should send these values to your Genesys representative.

You can also use the LINE Switcher API. This API allows you to transfer a conversation from LINE to a third-party Bot, then to Genesys Universal Messaging, and back to the third-party Bot. To use this feature, you will need to subscribe with LINE and a third-party bot.

Once this is done, you will need to write down several configuration values from the LINE Switcher API page and send them to your Genesys representative (along with the previous values).

For instance, here is a sample configuration:



You will need the **Switcher Service Code** and **Secret** from the Controller destination:

- Basic settings
- Messaging API
- LIFF
- Security
- Switcher API**
- Statistics
- Roles

### Destination

Refer and update information regarding each destination

Destination name ジェネシス- 【CCDemo】 GENESYS-CTL

Edit

Status

Destination ID 

Switcher secret 

And the **Switcher Destination Id** from the third-party Bot. That destination will be used by Universal Messaging to “switch” the customer when the conversation is done:

- Basic settings
- Messaging API
- LIFF
- Security
- Switcher API**
- Statistics
- Roles

### Destination

Refer and update information regarding each destination

Destination name DEMO endpoint 【CCDemo】 GENESYS

Edit

Status

Destination ID 

Once the configuration is done by Genesys, your Genesys representative will send you a webhook URL to configure in your LINE Business Account.

If you do not use the LINE Switcher API, you will need to configure the webhook URL in the **Messaging API** page:

### Webhook settings

Webhook URL ⓘ /line">https:///line 

Verify

Edit

If you use the LINE Switcher API, you will need to configure the webhook URL in the **Switcher Switcher** page under the destination that represents Genesys Universal Messaging:

Basic settings Messaging API LIFF Security Switcher API Statistics Roles

## Destination

Refer and update information regarding each destination

Destination name ジェネシス-【CCDemo】GENESYS-Manual

Edit

Status



Destination ID

[REDACTED]

Switcher secret

[REDACTED]

Webhook URL

https://[REDACTED]genesys[REDACTED]/line ←

Edit

## Using LINE's Call To LINE

Call To LINE is a feature from LINE allowing a third-party software, such as an IVR, to send messages to LINE users only by knowing their mobile phone numbers. If you have subscribed to this feature, you can use it with Universal Messaging.

First, you should request from your Genesys Representative the URL, Tenant Name or Id, and credentials to use with Universal Messaging.

Then, you can send messages with a POST Request following this format:

```
{
  "tenant": "yyy",
  "to": "telno",
  "messages": [
    { "type": "text", "text": "Hello!" },
    { "type": "sticker", "sticker_id": 12, "package_id": 1 }
    ...
  ]
}
```

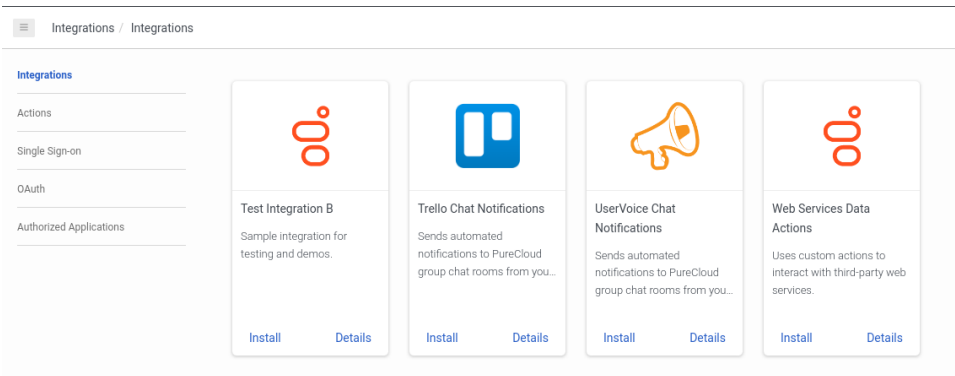
Where:

- `telno` is the LINE User's phone number, it has to be fully qualified (i.e. with the country code),
- `yyy` is either a Tenant Name or Id,
- `ddd` is the switcher destination, if not provided, the value from the Tenant in the config is used,
- `messages` are the messages to send to the LINE User.

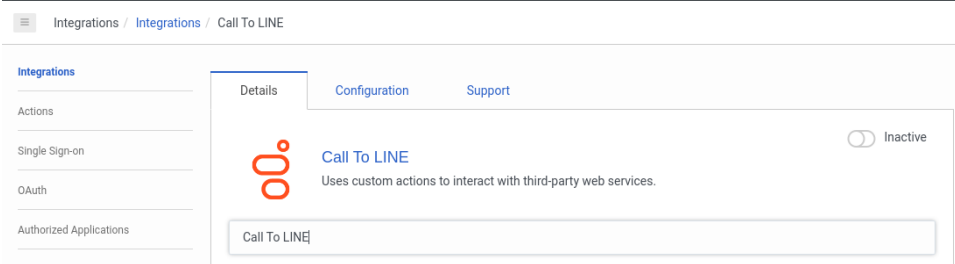
Notes:

- Each message must be a proper LINE message,
- Up to 5 messages at a time can be sent,
- Sending files is not yet supported.

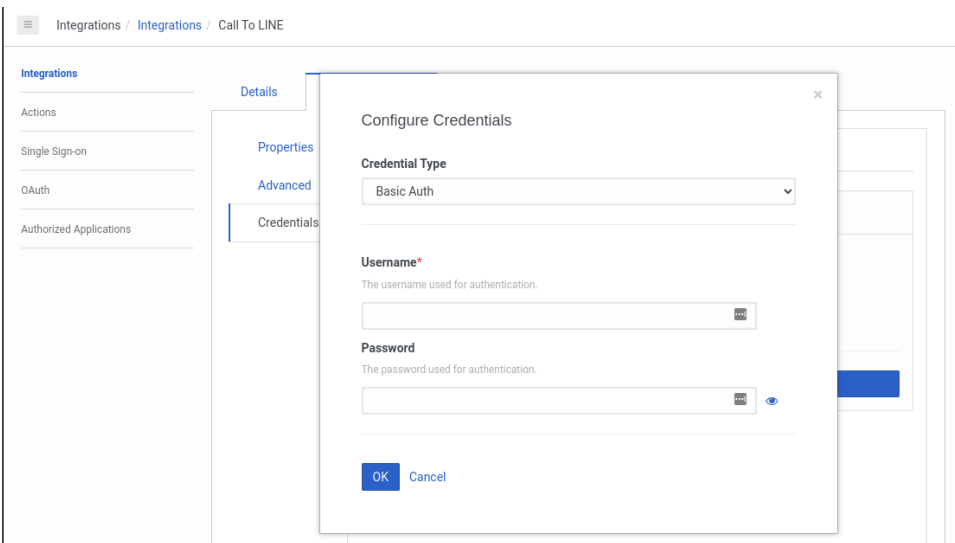
To use this feature from Genesys Cloud CX, just add a new `Web Services Data Actions` integration:



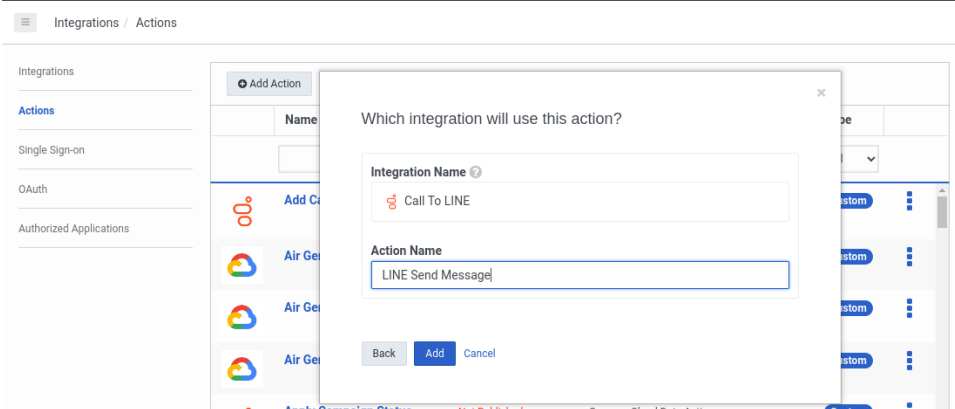
Give it a name:



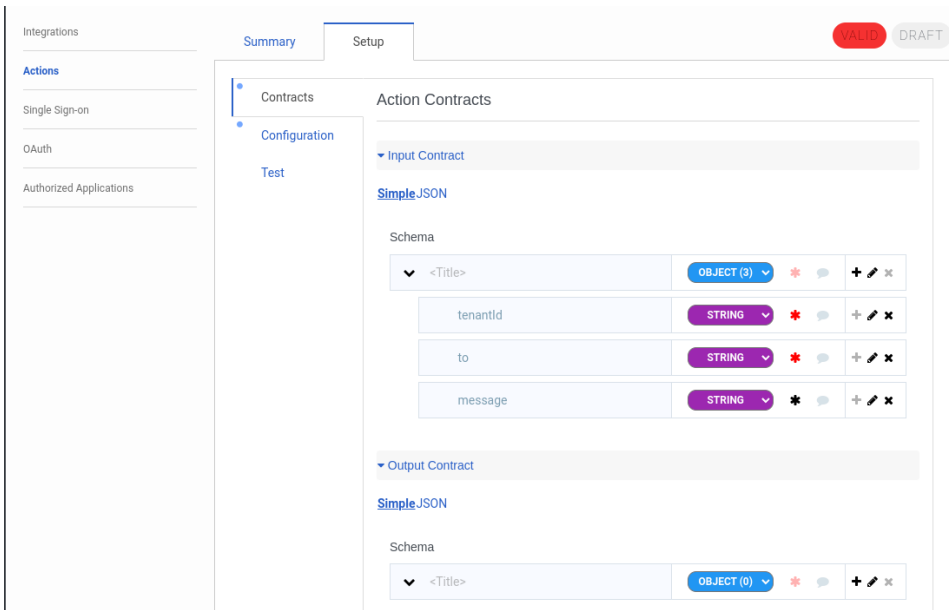
In the configuration tab, add the credentials:



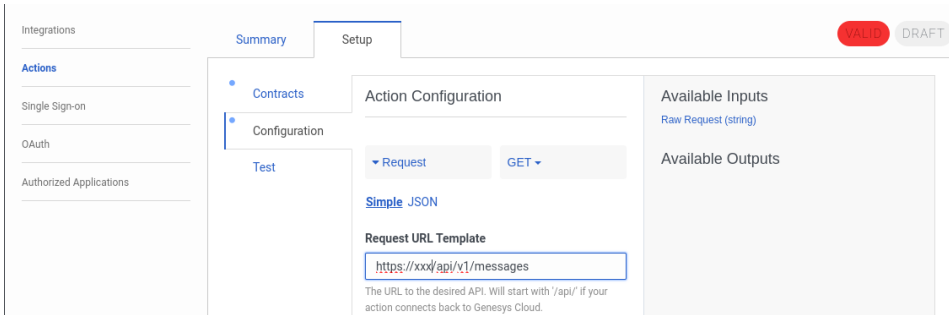
Then create a new Data Action :



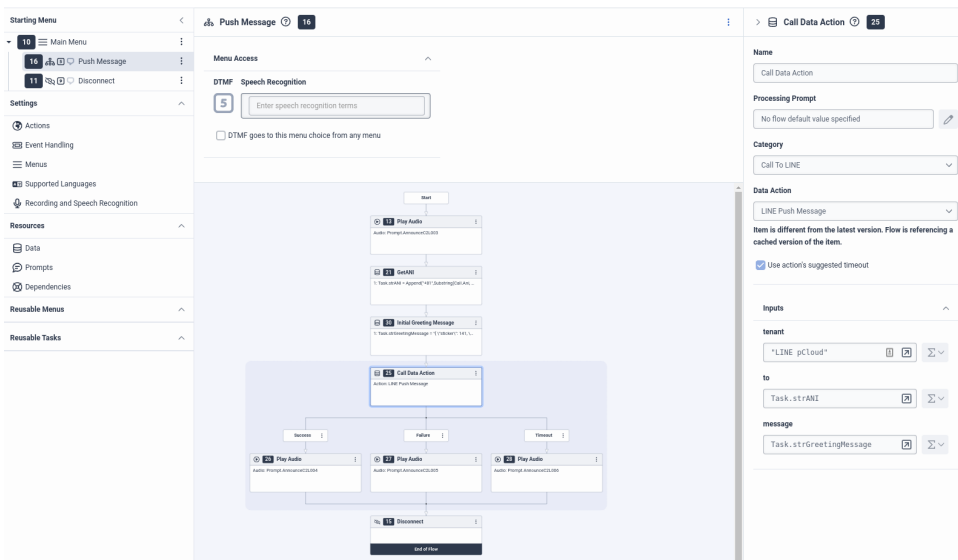
Define the Input and Output Contracts as follows:



And configure the URL (replace xxx with the URL provided by your Genesys representative):



Finally, you can use this action in an Architect Flow:



## Microsoft Teams

To configure Microsoft Teams, the customer must reach out the register link for Universal Messaging. Depending on the customer's region, the link will be different. The link is provided by the Universal Messaging team.

The first time the customer clicks on the link, they will be asked to fill in a form to register their Microsoft Teams organization:



Your Microsoft Teams Tenant is not yet registered with us

Please, fill in the form below to register it

**Business Name**

**Firstname**


**Lastname**


**Email**

**Microsoft Teams Tenant Identifier**

**Microsoft Teams Bot Name**

**Microsoft Teams Bot Accent Color**

**Microsoft Teams Bot Color Icon** (png only, 192x192 pixels)  
  

**Microsoft Teams Bot Outline Icon** (png only, 32x32 pixels, transparent or white)  
  

**Note:** If the customer sends several request to the register link, upon filling the form, they will see a message saying Genesys is already in the process of registering their organization. They will need to wait for the email from Genesys. If they are already registered, the message will give them the link to download the manifest file.

A Genesys representative will create a new tenant in Universal Messaging and contact the customer on the given email address. In that email there will be a link to download the manifest file to import in their Teams organization.

When the Universal Messaging Tenant is created, the Teams Connector will automatically associate the Teams Bot name, accent color, and icon to the tenant. The customer can change those values in the Universal Messaging Teams register page.

Success: Your Microsoft Teams Tenant is now registered with us


Please download the [Microsoft Teams Manifest](#) and import it to your Teams Organization. The manifest will define a new Microsoft Teams Bot named GUM-DEV the users can contact.


You can modify the following Teams Bot information (you will need to download the manifest again and re-install the Bot in your Teams Organization, however):

**Microsoft Teams Tenant Identifier**

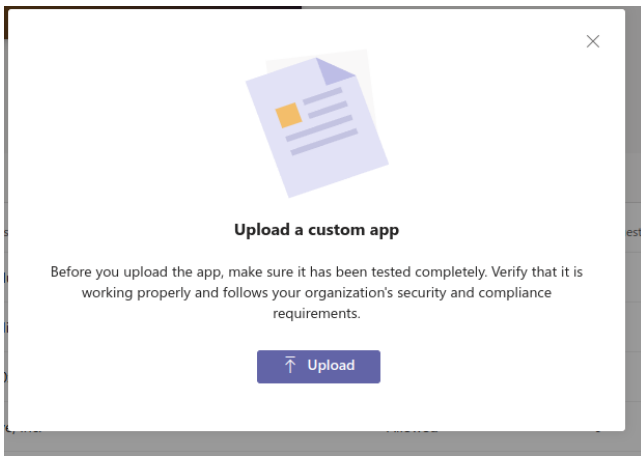
**Microsoft Teams Bot Name**

**Microsoft Teams Bot Accent Color**

**Microsoft Teams Bot Color Icon** (png only, 192x192 pixels)  
  

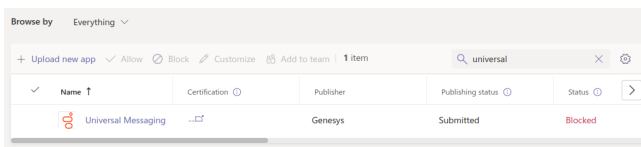
**Microsoft Teams Bot Outline Icon** (png only, 32x32 pixels, transparent or white)  
  

Modifying the Bot name, accent color, and icons does not require the customer to contact Genesys. However, if the manifest has been loaded already to Microsoft Teams, the customer will need to re-upload the manifest file to Microsoft Teams. The manifest version is automatically incremented when the customer changes the Bot name, accent color, and icons.

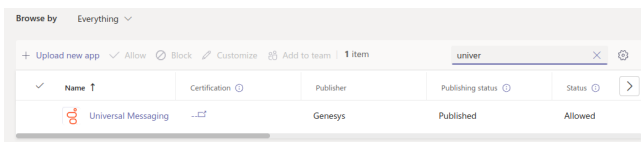


If the Teams App needs to be approved by a Teams Administrator. They need to go to [Microsoft Teams Admin Portal](#) and go to the "Manage Apps" section.

They should find the app with a "Submitted" Publishing status and a "Blocked" status:

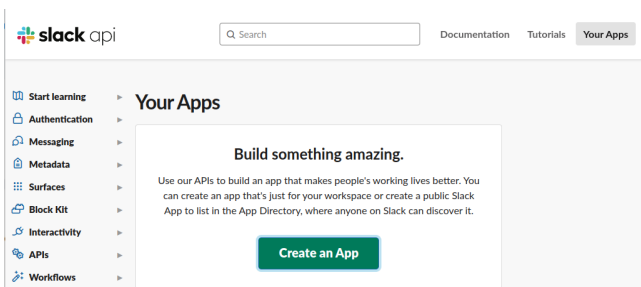


They should click on the App and then "Publish" it. After a few minutes, the app should be available in the Teams App Catalog:

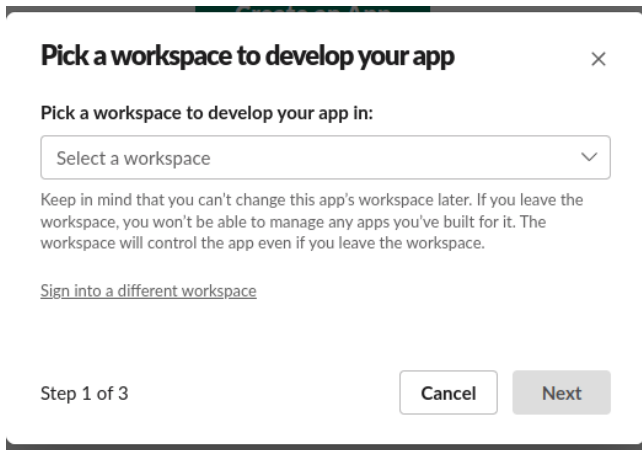


## Slack

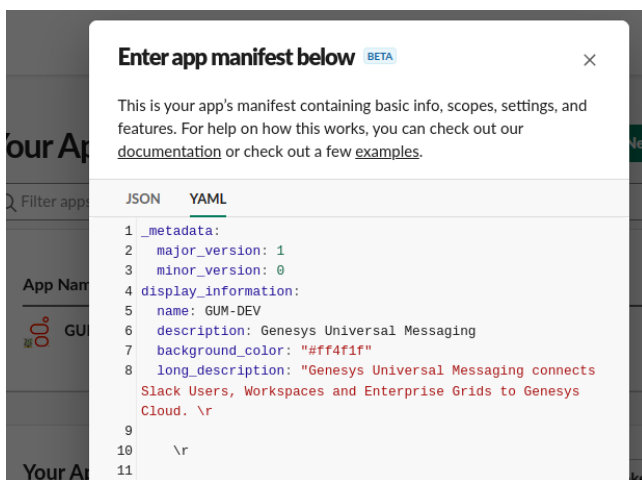
To configure Slack, the customer needs to create a new Slack App. They can do that by going to the [Slack App Directory](#) and clicking on the "Create New App" button:



And create the app from an app manifest, pick a workspace to create the app in:



Get the manifest from the [Universal Messaging Slack Manifest](#) website:

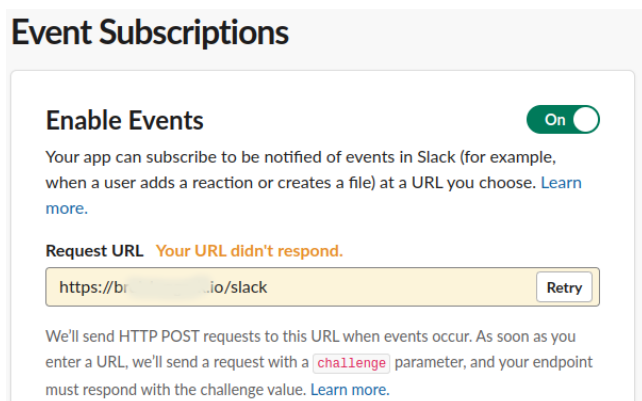


They can replace elements of the `display_information` in the dialog box, review the changes and click on `Create`. They can also modify the App Display information on the `Basic Information` and `App Home` pages.

The customer should copy the App ID, the Client ID, the Client Secret, and the Signing Secret from the Slack `Basic Information` page. And send them to their Genesys representative.

Once configured in Universal Messaging, the Genesys representative will send an identifier to the customer.

Once they receive that identifier, they can go to the `Event Subscriptions` page and hit the `Retry` button to validate the Request URL.



Finally, the customer should register their workspaces by going to `https://gum-xxx.com/slack/register`. The hostname of this URL is the same as the one in the manifest.

Your Slack Workspace is not yet registered with us

Please, fill in the form below to register it. You should have received your identifier from your Genesys Cloud representative

Customer Identifier

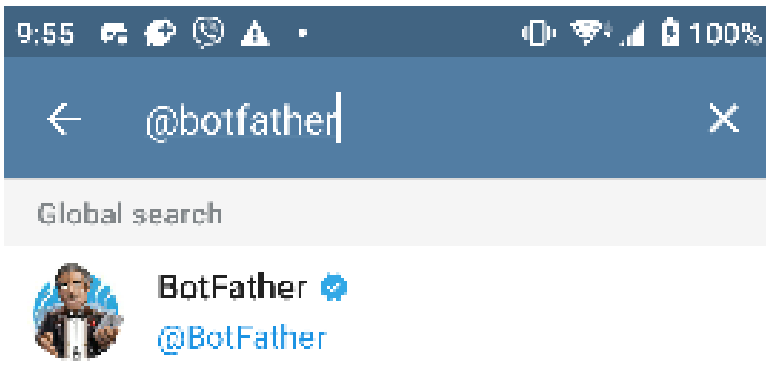
Submit

Once they approve the App in their workspace, they will be able to start conversations with Genesys Cloud agents.

## Telegram

To connect to Telegram, the customer needs to create a new Bot with Telegram's BotFather.

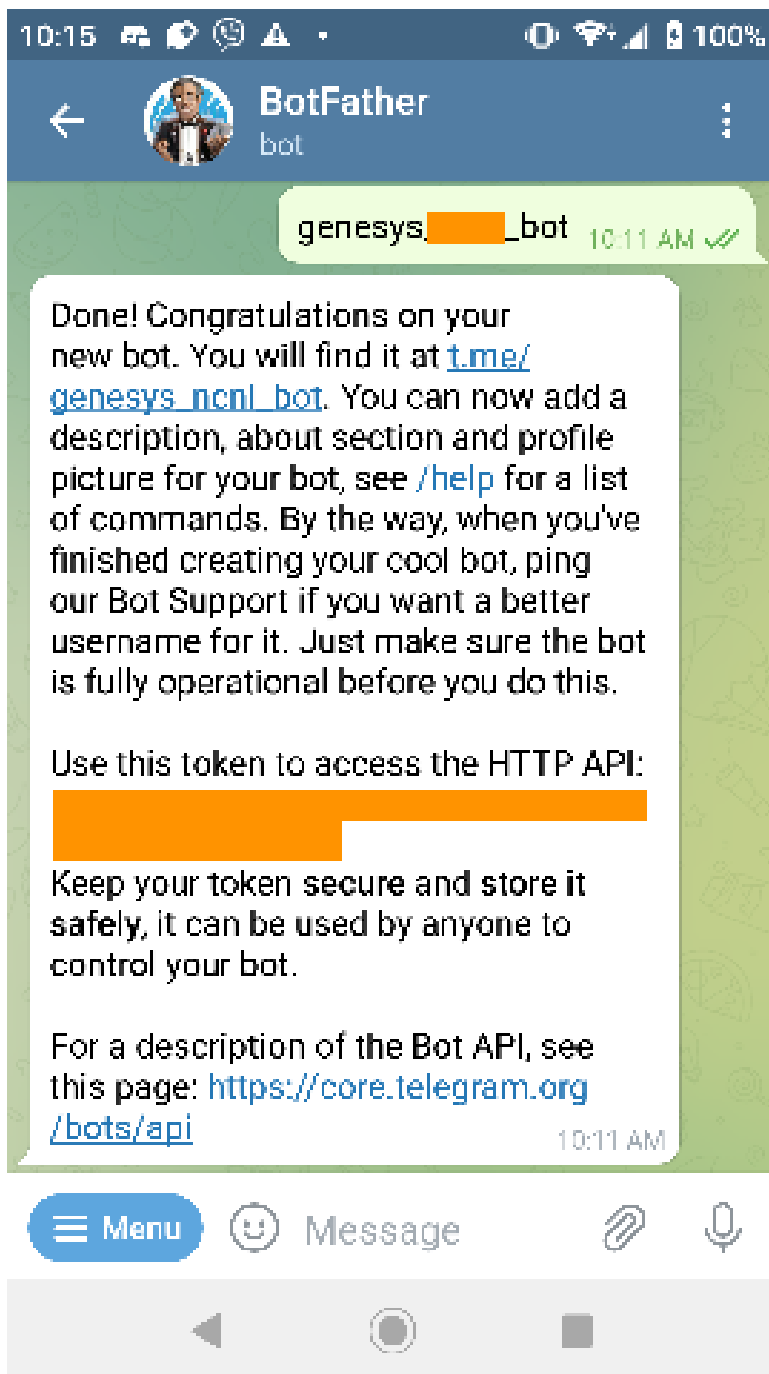
First, they need to chat with the BotFather by opening the search and typing @BotFather :



Once connected, they should start a new conversation (hit the **START** button) and create a new Bot by simply typing:

/newbot

This starts a dialog where they need to enter the name of your new Bot and its username. At the end of the dialog, the BotFather gives an API Token:



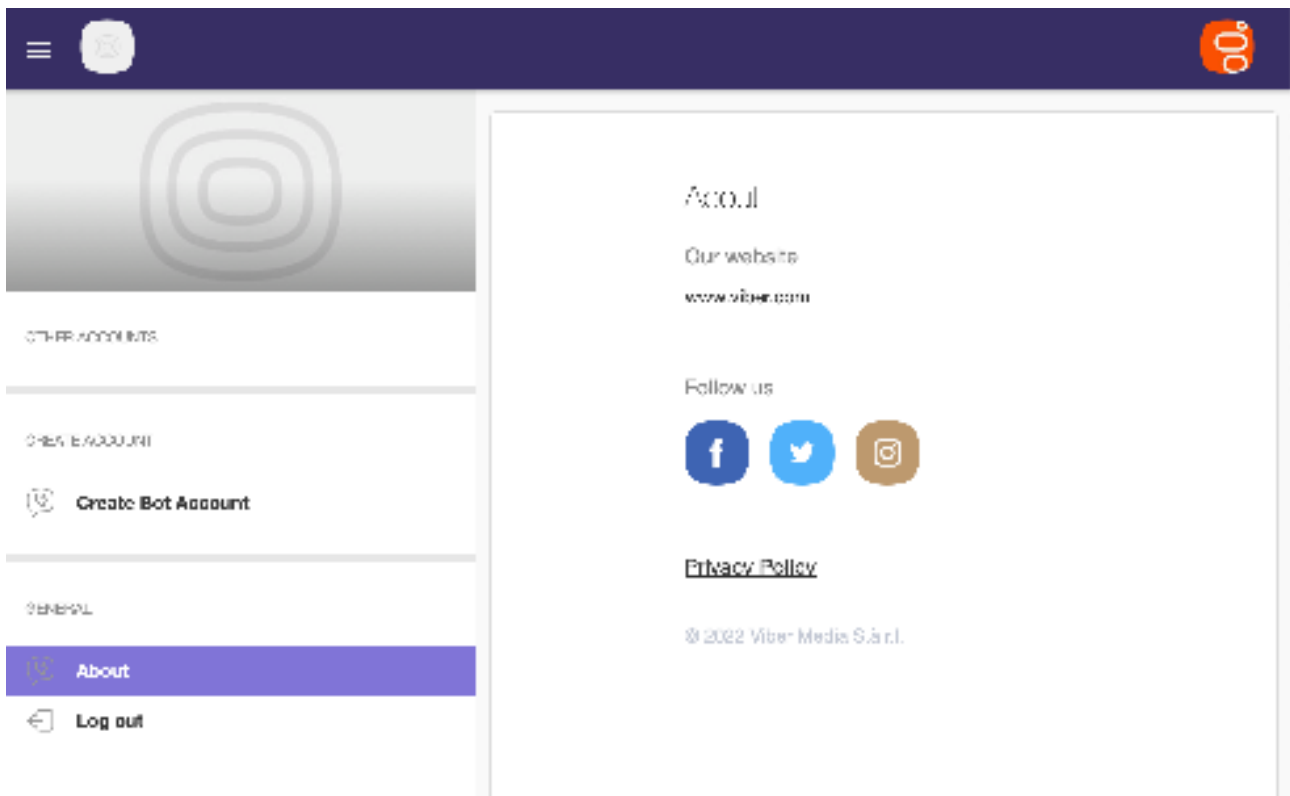
The customer needs to copy that API Token and send it to their Genesys representative.

End-users will start a conversation with the Bot by mentioning its username prepended with a @ .

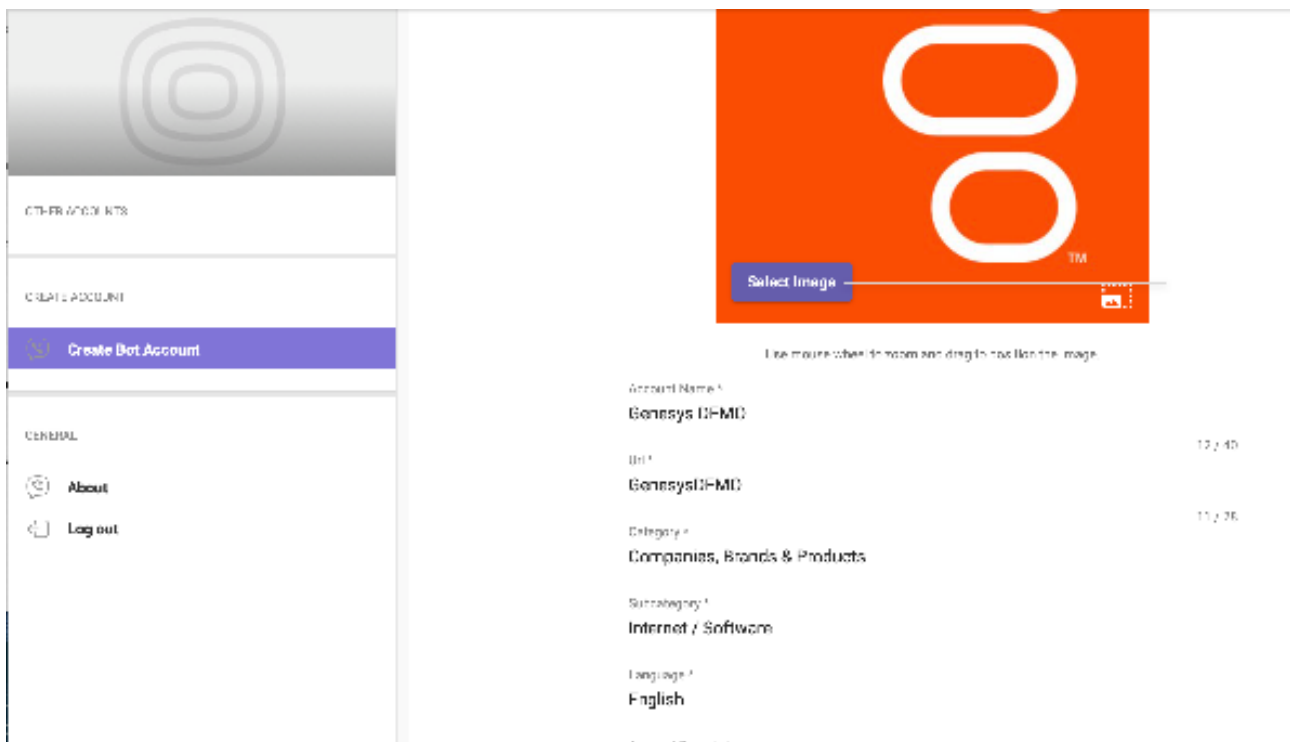
## Viber

The customer should create a [Viber Partner Account](#) and create a Viber Chatbot.

Then, log on your account at <http://partners.viber.com>:



And Create a Bot Account.



Upon creation, they will receive a Chat API Token and a URI that they need to send to their Genesys representative.

They can also send Genesys the name of the Viber Welcome Message and its text they want use with Viber when an end-user connects with their Viber Bot for the first time.

## Zalo

To connect to Zalo, the Zalo Official Account Administrator needs to connect to the Zalo console and [create a new Application](#).

Once the application is created, the administrator needs to register its usage with the Zalo Official Account API.

**Nộp các mục để duyệt**

Để có trải nghiệm tốt nhất khi tích hợp và sử dụng các API của Platform Zalo, bạn hãy lựa chọn các API cần dùng bên dưới và gửi ngay đến Zalo. Chúng tôi sẽ hỗ trợ bạn mở quyền đến các API mong muốn.

- ✓ Hãy lên danh sách các tác vụ cần thực hiện sau đó lựa chọn các API cần thiết trong menu. Điều này sẽ giúp tiết kiệm thời gian và không làm gián đoạn việc tích hợp và sử dụng API.
- ✗ Không xin quyền sử dụng tất cả API cùng một lúc. Việc này sẽ khiến thời gian xét duyệt tăng lên khiến bạn phải chờ đợi.

**Gửi xét duyệt sử dụng Official Account API**

API	Số lần đã gọi/Giới hạn (Theo ngày)	Số lần đã gọi/Giới hạn (Theo tuần)	Số lần đã gọi/Giới hạn (Theo tháng)	Trạng thái
Cập nhật thông tin của người quan tâm	0 / không giới hạn	0 / không giới hạn	0 / không giới hạn	Đã được duyệt
Lấy danh sách người quan tâm	0 / không giới hạn	0 / không giới hạn	0 / không giới hạn	Đã được duyệt
Lấy thông tin người quan tâm	2 / không giới hạn	2 / không giới hạn	49 / không giới hạn	Đã được duyệt
Xóa nhân	0 / không giới hạn	0 / không giới hạn	0 / không giới hạn	Đã được duyệt
Gỡ nhân khỏi người quan tâm	0 / không giới hạn	0 / không giới hạn	0 / không giới hạn	Đã được duyệt
Lấy thông tin nhân	0 / không giới hạn	0 / không giới hạn	0 / không giới hạn	Đã được duyệt

[Nộp xét duyệt](#)

Then the administrator needs to link the Zalo Official Account to this application (that account needs to exist already and require some legal documents to be submitted to Zalo):

**Official Account**

Official Account cho phép bạn tạo một công đồng người dùng quan tâm mà tại đó họ có thể tương tác trực tiếp với Official Account, cập nhật và trao đổi các thông tin liên quan đến nhau cầu thủ trực tiếp.

**Liên kết với Official Account**

Việc liên kết sẽ kích hoạt trả về thêm trường `user_id_by_app` ở event webhook, `user_id_by_app` này sẽ tương ứng với id của user khi đăng nhập vào ứng dụng của bạn.

Official Account  [Liên kết](#)

Bạn chưa có Official Account. [Click vào đây để tạo Official Account mới](#)

Once this is done, they should send to their Genesys representative the following information:

- Application Identifier,
- Application Secret Key,
- Official Account (OA) Identifier,
- Official Account (OA) Secret Key,
- Domain Verification Meta tag

Next, the administrator needs to verify the domain used by Universal Messaging by entering the webhook from Universal Messaging Config in the `Webhook` field.

**Xác thực domain**

Bạn cần xác minh domain của mình - địa chỉ website của bạn (ví dụ: example.com, zalo\_verifier.example.com) trước khi bạn sử dụng cho các dịch vụ của Zalo Platform. Điều này giúp bạn đảm bảo không có ai khác đang sử dụng domain của bạn mà không có sự cho phép của bạn.

Bạn có thể xem hướng dẫn chi tiết [tại đây](#) hoặc bắt đầu nhanh ngay bên dưới.

Domain  [Xác thực](#)

Tiền tố URL  [Xác thực](#)

The domain verification string, which is the content of the meta tag `zalo-platform-site-verification`, will be needed in the Universal Messaging configuration.

### Xác thực quyền sở hữu

URL : <https://acme.com/zalo/>

Chọn cách xác thực

- Tải tệp HTML lên trang web của bạn >
- Thêm thẻ meta vào trang web của bạn v

**Bước 1:** Sao chép thẻ meta sau:

```
<meta name="zalo-platform-site-verification" content="IyxX6uLH.../rqssZGqICZS" />
```

**Bước 2:** Dán thẻ meta trên vào mục head của trang web: <https://acme.com/zalo/>  
**Chú ý:** Đưa các thẻ meta lên đầu trang (chúng ở quá xa và vượt quá kích thước cho phép thu thập từ Zalo (512kb) thì sẽ không lấy được). Bỏ chặn request từ các IP nước ngoài. Tăng tốc độ load lần đầu của trang web.

**Bước 3:** Thực hiện publish trang web của bạn, hãy xác nhận rằng thẻ meta hiển thị bằng cách truy cập vào trang web và xem nguồn HTML.

**Bước 4:** Chọn vào nút **Xác thực** bên dưới.  
**Chú ý:** Có thể mất tới 72 giờ để Zalo Platform tìm thấy thẻ meta. Nếu trạng thái domain vẫn là **Chưa được xác thực**, bạn có thể sử dụng công cụ [Gỡ lỗi chia sẻ](#) để kiểm tra thẻ meta của bạn.

[Hướng dẫn chi tiết](#)

Once the domain is verified, Genesys will send the authorization link to the Zalo Official Account administrator for validation.

Follow the instructions about configuring the webhook in the webhook section of the same page. The Zalo Official Account administrator needs to enter the webhook from Universal Messaging Config in the **Webhook** page on Zalo and give the listed permissions.





## Chapter 3

# Architect Inbound Flows

To process conversations, you will need to write a Genesys Architect Inbound Message Flow. While explaining how to write such a flow is out of the scope of this document, there is some information Genesys Universal Messaging will provide to help you in such a task.

When the conversation reaches your Architect flow, the Participant Data of the end-customer will contain additional entries that will allow the flow to make appropriate decisions on how to proceed (sending to a queue, starting a Bot, etc):

- `media` ,  
The name of the Messaging Connector, one of:  
Apple Messages for Business, Bedore, BizM KakaoTalk, Disqus, Google Business Messages, Google Chat, Infobank KakaoTalk, LINE, Media4U, Slack, Teams, Telegram, Viber, WeChat, Zalo
- `capabilities` ,  
The comma-separated list of capabilities supported by the guest device as provided by the Social Media, [Apple Messages for Business](#)
- `country` ,  
The country where the guest is located,  
Viber, WeChat
- `city` ,  
The city where the guest is located,  
WeChat
- `deviceOS` ,  
The OS of the device of the guest, Viber
- `deviceType` ,  
The device type of the guest, Viber
- `enterpriseId` ,  
The Slack Enterprise Identifier, if applicable,  
Slack
- `enterpriseName` ,  
The Slack Enterprise Name, if applicable,  
Slack
- `forum` ,  
The forum where the conversation was started,  
Disqus
- `googleAgent`  
The Google Agent that was used, Google Business Messages
- `googleConversationId`  
The Google Conversation ID, Google Business Messages
- `googleEntryPoint`  
The Google Entry Point if present,  
Google Business Messages
- `googleGTIN`  
The Google Shopping Global Trade Item Number (GTIN) if present,  
Google Business Messages
- `googleMerchantOfferId`  
The Google Shopping Merchant Id if present, Google Business Messages
- `googleNearPlaceId`

- The Google Near Place Identifier if present,  
Google Business Messages
- `googlePlaceId`  
The Google Place Identifier if present,  
Google Business Messages
- `googleProductCategory`  
The Google Shopping Product Category if present,  
Google Business Messages
- `googleProductImageURI`  
The Google Shopping Product Image URI if present,  
Google Business Messages
- `googleProductTitle`  
The Google Shopping Product Title if present,  
Google Business Messages
- `googleProductURI`  
The Google Shopping Product URI if present,  
Google Business Messages
- `group` ,  
The group identifier as provided by the Social Media.  
[Apple Messages for Business](#)
- `intent` ,  
The Intent that started this conversation (if any).  
[Apple Messages for Business](#)
- `locale`  
[Apple Messages for Business](#), Google Business Messages, Viber, WeChat
- `province` ,  
The province where the guest is located,  
WeChat
- `subscribeScene` ,  
The place where the guest is,  
WeChat
- `thread` ,  
The thread where the conversation was started,  
Disqus
- `workspaceId` ,  
The Slack Workspace Identifier,  
Slack
- `workspaceName` ,  
The Slack Workspace Name,  
Slack

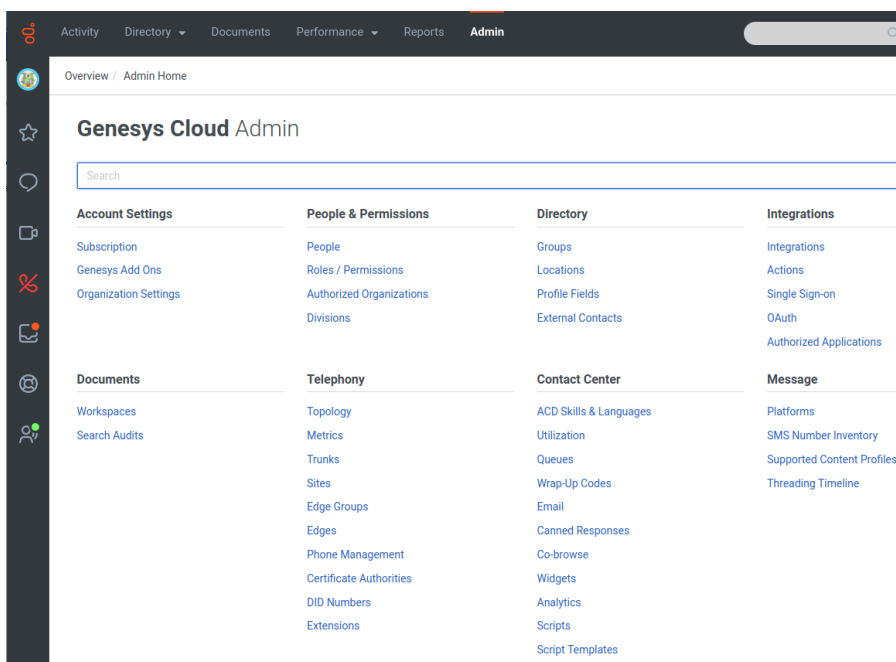
# Chapter 4

## Canned Responses

Canned Responses are used in Universal Messaging to send predefined messages to social media users. These messages can be simple text or complex JSON objects that are either common to Genesys Cloud or specific to each social media.

### Create a Canned Response

To add a canned response for Universal Messaging, you must create a canned response in Genesys Cloud, in the **Admin/Canned Responses** section:



You can add or reuse a Response Library as you see fit. A Response Library is simply a collection of canned responses. When adding a Response, make sure you choose **Campaign SMS Template** as the Response Type. This is the only type supported by Universal Messaging (if you use another type, you will get an error when trying to send the message):

**Response Name**

**Library**

**Response Type**

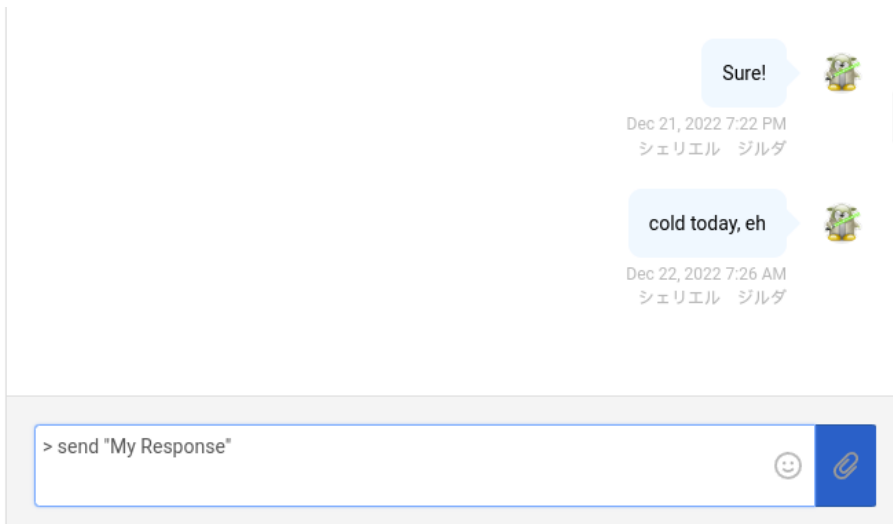
Standard Message Template **Campaign SMS Template** Campaign Email Template Footer

Some great response!

## Send a Canned Response

Once created, you can send the canned response from a Bot or an agent by sending the following command to Universal Messaging:

```
> send "My Response"
```

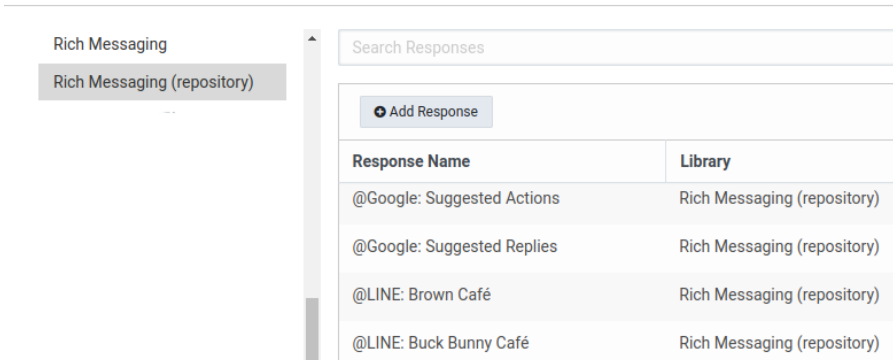


You can even create a Canned Response for sending another Canned Response, so you do not have to type '> send "My Response"' every time.

At Genesys, we typically name the original Canned Response `@My Response` and save it in a Response Library called `XXX (repository)` and then create a new Canned Response called `My Response` in a Response Library called `XXX`.

As shown here:

[Responses / Manage Libraries](#)



In `My Response`, we simply add the following text:

```
> send "@My Response"
```

## Complex Canned Responses

Sending a complex object (Cards, Carrousels, etc) will be done similarly, as follows:

**Response Name**

**Library**

**Response Type**

Standard
  Message Template
  Campaign SMS Template
  Ca

```

{
  "data": {
    "quick-reply": {
      "summaryText": "What can I help you with?",
      "items": [
        {
          "identifier": "1",
          "title": "Watch our video"
        },
        {
          "identifier": "2",
          "title": "Ask for office hours"
        },
        {
          "identifier": "3",
          "title": "Find more about pet adoption"
        },
        {
          "identifier": "4",
          "title": "Talk to an agent"
        }
      ]
    }
  }
}

```

Social Media have different ways of encoding the JSON object. For example, the Quick Reply is encoded by Apple Messages for Business as follows:

```

{
  "data": {
    "quick-reply": {
      "summaryText": "What can I help you with?",
      "items": [
        {
          "identifier": "1",
          "title": "Watch our video"
        },
        {
          "identifier": "2",
          "title": "Ask for office hours"
        }
      ]
    }
  }
}

```

```

        "identifier": "3",
        "title": "Find more about pet adoption"
    },
    {
        "identifier": "4",
        "title": "Talk to an agent"
    }
]
}
}
}

```

While Google Business Messaging would be like:

```

{
  "text": "Hello, world!",
  "fallback": "Hello, world!\n\nReply with \"Hello\" or \"Hi!\"",
  "suggestions": [
    {
      "reply": {
        "text": "Hello",
        "postbackData": "hello-formal"
      }
    },
    {
      "reply": {
        "text": "Hi!",
        "postbackData": "hello-informal"
      }
    }
  ]
}

```

General information about the JSON objects can be found here:

- Genesys
  - Work with Quick Replies in Bot Conversations
  - Work with Cards in Bot Conversations (upcoming)
  - Work with Carousels in Bot Conversations (upcoming)
  - Universal Messages will transform the Genesys objects into the appropriate JSON object for each social media.
- Apple Messages for Business
  - Rich Link Messages
  - Interactive Messages
  - Notes:** You typically need the `data` object in the Canned Response (as shown earlier). Universal Messaging will add the `bid`, `destinationId`, `id`, `requestIdentifier`, `sourceId`, `version` automatically.
- Google Chat for Workspaces
  - Cards
  - Google has a [Card builder](https://goo.gle/card-builder) to create the JSON object: <https://goo.gle/card-builder>
  - Once built in their editor, simply copy and paste the JSON in the Canned Response.
  - Universal Messages will bundle the card in a `cardsV2` array and give it a `cardId`.
  - To start a Dialog from a Card, you should store the Dialog in a Canned Response, and have the card referencing it:

```

"buttons": [
  {
    "text": "Edit",
    "onClick": {
      "action": {
        "interaction": "OPEN_DIALOG",
        "function": "@GChat: Edit Contact Dialog", // the name of the Canned Response
        "parameters": [
          {

```

```

        "key": "type",
        "value": "cannedResponse" // Tells Universal Messaging to fetch
                                   // a Canned Response
    },
    {
        "key": "id",
        "value": "sasha"
    }
]
}
},
...
]

```

- Google Business Messages
  - [Send Messages](#)
  - [Message Reference API](#)
  - Universal Messages will automatically add the `messageId` for you.
  
- LINE
  - [Flex Messages](#)
  - While the other formats are also supported, Flex Messages are the most powerful and flexible. We strongly encourage you to use them over their older formats (they are supported by Universal Messages anyway) LINE provides a [Flex Message Builder](#) to create the JSON object: <https://developers.line.biz/flex-simulator>. **Note:** Once the message is built, get the JSON object (via the [View as JSON](#) button) and embed it with `{ "type": "flex", "altText": "title", "contents": ... }` before adding it to the Canned Response (See: [Call the Messaging API to send a Flex Message](#)).
  
- Microsoft Teams
  - [Microsoft Adaptive Cards](#)
  - Microsoft also provides an [Adaptive Card Designer](#) to create the JSON object: <https://adaptivecards.io/designer>
  
- Slack
  - [Composing Layouts](#)
  - Slack also provides a Card builder, called the [Block Kit Builder](#): <https://app.slack.com/block-kit-builder>
  
- Telegram
  - Telegram Buttons are used via Genesys Cloud Quick Replies.
  
- Viber
  - [Viber Carousels](#)
  - [Keyboards](#)
  - [Stickers](#)
  - Universal Messages will add the `receiver` , `sender` , and `min_api_version` fields automatically.
  
- Zalo
  - [Gửi thông báo theo mẫu đính kèm danh sách](#)
  - [Gửi thông báo kèm Sticker](#)
  - [Cấu trúc của tham số buttons](#)

Have a look at the [Samples](#) section for more details.

## Templated Canned Responses

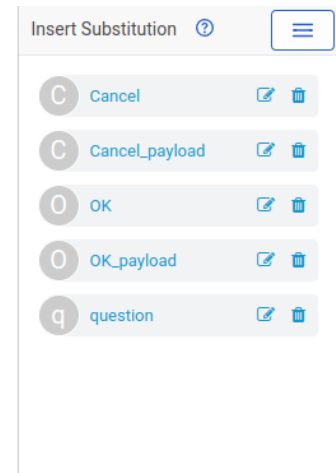
It is also possible to use placeholders and complex expressions in Canned Responses. The placeholder can be configured as you would in a Genesys Cloud SMS Campaign, also called “Substitutions”. Here is a sample:



```

{
  "genesys_prompt": "{{question}}",
  "genesys_quick_replies": [{
    {{- if .OK_payload}}
      "text": "{{(OK)}}", "payload": "{{(OK_payload)}}"
    {{- else}}
      "text": "{{(OK)}}", "payload": "answer={{(OK)}}"
    {{- end}}
  ]},
  {{- if .Cancel_payload}}
    "text": "{{(Cancel)}}", "payload": "{{(Cancel_payload)}}"
  {{- else}}
    "text": "{{(Cancel)}}", "payload": "answer={{(Cancel)}}"
  {{- end}}
  ]}
}

```



You can see the substitution `question` is mentioned in the JSON as `{{question}}`. You can set its default value in Genesys Cloud:

**Name**

**Description**

The question to ask the customer

**Default Value**

The Go Template syntax can also be used to create more complex expressions. Here is an example:

```

{
  "name": "{{name}}", // GC placeholder
  {{ if gt .total 10}} // Go Template
    "text": "{{.name}}, You are eligible for a discount", // Go Template Placeholder
  {{ else }}
    "text": "Next time maybe",
  {{ end }}
  "type": "Button"
}

```

You can also use the [Sprig Template Library](#):

```
{
  "text": "{{default "Hello" .name}}"
}
```

Then, from a Bot or an agent you can call the Canned Response with parameters:

send "cannedresponse" with name="John" total="12"

If you do not provide a value for a placeholder, the default value will be used, from Genesys Cloud.

## Samples

Here we will show a non-exhaustive list of samples from social media documentations and how they should look like in a Canned Response.

In these samples, when the line is too long for the documentation, we cut it with the `↵` character. You should not cut them when you create your Canned Responses.

### Genesys

Location (does not work with all social media):

```
{
  "type": "location",
  "title": "Genesys HQ",
  "address": "2001 Junipero Serra Boulevard Daly City, California 94014, United States",
  "latitude": 37.7011118,
  "longitude": -122.4719997
}
```

Quick Replies (using templates):

```
{
  "genesys_prompt": "{{question}}",
  "genesys_quick_replies": [{
    {{- if .OK_payload}}
      "text": "{{OK}}", "payload": "{{OK_payload}}"
    {{- else}}
      "text": "{{OK}}", "payload": "answer={{OK}}"
    {{- end}}
  ], {
    {{- if .Cancel_payload}}
      "text": "{{Cancel}}", "payload": "{{Cancel_payload}}"
    {{- else}}
      "text": "{{Cancel}}", "payload": "answer={{Cancel}}"
    {{- end}}
  ]
}
```

Stickers (the sticker Identifier must mean something on the target Social Media):

```
{
  "type": "sticker",
  "title": "Hug",
  "stickerId": 40130
}
```

This is a "Hug" sticker in Viber, for example.

### Apple Messages for Business

Payment Request (Note: For this to work, you need to set up a Payment Provider with Apple, Genesys cannot provide this for you, and send its configuration to your Genesys representative):

```
{
  "data": {
    "images": [
      {
        "url": "https://picsum.photos/250",
        "identifier": "1"
      }
    ],
    "payment": {
      "paymentRequest": {
        "countryCode": "US",
        "currencyCode": "USD",
        "lineItems": [
          {
            "amount": "1.00",
            "label": "Stickers",
            "type": "final"
          },
          {
            "amount": "0.50",
            "label": "Shipping",
            "type": "final"
          }
        ],
        "requiredBillingContactFields": [
        ],
        "requiredShippingContactFields": [
        ],
        "shippingMethods": [
          {
            "amount": "0.00",
            "detail": "Available within an hour",
            "identifier": "in_store_pickup",
            "label": "In-Store Pickup"
          },
          {
            "amount": "0.50",
            "detail": "5-8 Business Days",
            "identifier": "flat_rate_shipping_id_2",
            "label": "UPS Ground"
          },
          {
            "amount": "29.99",
            "detail": "1-3 Business Days",
            "identifier": "flat_rate_shipping_id_1",
            "label": "FedEx Priority Mail"
          }
        ],
        "total": {
          "amount": "1.50",
          "label": "Genesys Test",
          "type": "final"
        }
      }
    }
  },
  "receivedMessage": {
    "imageIdentifier": "1",
    "style": "large",
    "subtitle": "$1.50 at Genesys Test",
  }
}
```

```

    "title": "Stickers"
  }
}

```

Authentication Request (Note: for this to work, you need to set up an Authentication Provider with Apple, Genesys cannot provide this for you, and send its configuration to your Genesys representative):

```

{
  "data": {
    "images": [
      {
        "url": "https://avatars.githubusercontent.com/u/2824157?s=200&v=4",
        "identifier": "1"
      }
    ],
    "authenticate": {}
  },
  "receivedMessage": {
    "imageIdentifier": "1",
    "style": "large",
    "title": "Sign in to Auth0"
  },
  "replyMessage": {
    "title": "You are now signed in"
  }
}

```

List Picker:

```

{
  "data": {
    "listPicker": {
      "sections": [
        {
          "items": [
            {
              "identifier": "1",
              "order": 0,
              "style": "default",
              "subtitle": "Red and delicious",
              "title": "Apple"
            },
            {
              "identifier": "2",
              "order": 1,
              "style": "default",
              "subtitle": "Vitamin C boost",
              "title": "Orange"
            }
          ],
          "order": 0,
          "title": "Fruit",
          "multipleSelection": true
        },
        {
          "items": [
            {
              "identifier": "3",
              "order": 0,
              "style": "default",
              "subtitle": "Crispy greens",
              "title": "Lettuce"
            }
          ]
        }
      ]
    }
  }
}

```

```

    },
    {
      "identifier": "4",
      "order": 1,
      "style": "default",
      "subtitle": "Not just for your eye lids",
      "title": "Cucumber"
    }
  ],
  "order": 1,
  "title": "Veggies"
}
]
}
}
}

```

## Rich Link:

```

{
  "body": "https://www.apple.com/ipad-pro/",
  "type": "richLink",
  "richLinkData": {
    "url": "https://www.apple.com/ipad-pro/",
    "title": "iPad Pro",
    "assets": {
      "image": {
        "url": "
https://support.apple.com/library/content/dam/edam/applecare/images/en_US/
ipad/ipadpro/2021-ipad-pro-12-9-colors.png",
        "mimeType": "image/jpeg"
      }
    }
  }
}
}

```

## Time Picker:

```

{
  "data": {
    "event": {
      "identifier": "1",
      "title": "Please Pick a date for your dentist appointment",
      "location": {
        "title": "Genesys Japan",
        "latitude": 35.6650161,
        "longitude": 139.7432184,
        "radius": 17
      }
    },
    "timezoneOffset": 5400,
    "timeslots": [
      {
        "identifier": "0",
        "startTime": "2022-07-10T17:00+0000",
        "duration": 3600
      },
      {
        "identifier": "1",
        "startTime": "2022-07-31T15:00+0000",
        "duration": 3600
      }
    ]
  }
}

```

```

        "identifier": "2",
        "startTime": "2022-08-12T14:00+0000",
        "duration": 3600
    }
]
},
"receivedMessage": {
    "imageIdentifier": "0",
    "style": "icon",
    "title": "Please pick a time",
    "subtitle": "This should easy"
},
"replyMessage": {
    "style": "icon",
    "title": "Thank you!"
}
}

```

It is also possible to use third-party applications. In that case you will need to fill in the `bid` of the application, for example. Due to the complexity of these, it would be better for you to be in contact your Genesys representative:

```

{
    "bid": "com.apple.messages.MSMessageExtensionBalloonPlugin:U9S8EQXSA2:
com.example.apple-samplecode.PackageDeliveryU9S8EQXSA2.MessagesExtension",
    "appId": "123456789",
    "appName": "Package Delivery",
    "URL": "?name=WWDC%20Goodies&deliveryDate=09-06-2017&
destinationName=Moscone%20Convention%20Center&street=747%20Howard%20St&
state=CA&city=San%20Francisco&country=USA&postalCode=94103&
latitude=37%2E7831&longitude=%2D122%2E4041&extraCharge=15%2E00",
    "receivedMessage": {
        "title": "WWDC Goodies",
        "subtitle": "Scheduled for delivery - Today"
    },
    "useLiveLayout": true
}

```

## Google Chat for Workspaces

Card:

```

{
    "header": {
        "title": "Sasha",
        "subtitle": "Software Engineer",
        "imageUrl": "https://developers.google.com/chat/images/quickstart-app-avatar.png",
        "imageType": "CIRCLE",
        "imageAltText": "Avatar for Sasha"
    },
    "sections": [
        {
            "header": "Contact Info",
            "collapsible": true,
            "uncollapsibleWidgetsCount": 1,
            "widgets": [
                {
                    "decoratedText": {
                        "startIcon": {
                            "imageType": "CIRCLE",
                            "knownIcon": "EMAIL"
                        }
                    }
                }
            ]
        }
    ]
}

```

```

    },
    "text": "sasha@example.com"
  }
},
{
  "decoratedText": {
    "startIcon": {
      "imageType": "CIRCLE",
      "knownIcon": "PERSON"
    },
    "text": "<font color=#80e27e>Online</font>"
  }
},
{
  "decoratedText": {
    "startIcon": {
      "imageType": "CIRCLE",
      "knownIcon": "PHONE"
    },
    "text": "+1 (555) 555-1234"
  }
},
{
  "buttonList": {
    "buttons": [
      {
        "text": "Share",
        "onClick": {
          "openLink": {
            "url": "https://example.com/share"
          }
        }
      },
      {
        "text": "Edit",
        "onClick": {
          "action": {
            "interaction": "OPEN_DIALOG",
            "function": "@GChat: Edit Contact Dialog",
            "parameters": [
              {
                "key": "type",
                "value": "cannedResponse"
              },
              {
                "key": "id",
                "value": "sasha"
              }
            ]
          }
        }
      }
    ]
  }
},
{
  "text": "Print",
  "onClick": {
    "action": {
      "function": "print",
      "parameters": [
        {
          "key": "type",

```

```
        "value": "PDF"
      }
    ]
  }
}
]
```

Dialog (initiated by the previous card):

```
{
  "sections": [
    {
      "header": "Edit Contact",
      "widgets": [
        {
          "textInput": {
            "type": "SINGLE_LINE",
            "name": "name",
            "label": "Name"
          }
        },
        {
          "textInput": {
            "type": "MULTIPLE_LINE",
            "name": "address",
            "label": "Address"
          }
        },
        {
          "dateTimePicker": {
            "type": "DATE_AND_TIME",
            "name": "datetime",
            "label": "Pick a date and time"
          }
        },
        {
          "decoratedText": {
            "text": "Add to favorites",
            "switchControl": {
              "controlType": "SWITCH",
              "name": "saveFavorite"
            }
          }
        },
        {
          "decoratedText": {
            "text": "Merge with existing contacts",
            "switchControl": {
              "controlType": "SWITCH",
              "name": "mergeContact",
              "selected": true
            }
          }
        }
      ]
    }
  ],
}
```



```

{
  "selectionInput": {
    "type": "RADIO_BUTTON",
    "label": "Contact type",
    "name": "contactType",
    "items": [
      {
        "text": "Work",
        "value": "Work",
        "selected": false
      },
      {
        "text": "Personal",
        "value": "Personal",
        "selected": false
      }
    ]
  },
  "buttonList": {
    "buttons": [
      {
        "text": "Submit",
        "color": {
          "red": 0.4,
          "green": 0.717,
          "blue": 0.227,
          "alpha": 0
        },
        "onClick": {
          "action": {
            "function": "saveContact"
          }
        }
      }
    ]
  }
}

```

## Google Business Messages

Suggested Replies:

```

{
  "text": "Hello, world!",
  "fallback": "Hello, world!\n\nReply with \"Hello\" or \"Hi!\"",
  "suggestions": [
    {
      "reply": {
        "text": "Hello",
        "postbackData": "hello-formal"
      }
    },
    {
      "reply": {
        "text": "Hi!",

```

```

    "postbackData": "hello-informal"
  }
}
]
}

```

Card:

```

{
  "fallback": "Hello, world!\nReply with \"Suggestion #1\" or \"Suggestion #2\"",
  "richCard": {
    "standaloneCard": {
      "cardContent": {
        "title": "Hello, world!",
        "description": "Sent with Business Messages.",
        "media": {
          "height": "TALL",
          "contentInfo": {
            "altText": "Google logo",
            "fileUrl": "
https://www.google.com/images/branding/googlelogo/2x/googlelogo_color_272x92dp.png",
            "forceRefresh": false
          }
        },
        "suggestions": [
          {
            "reply": {
              "text": "Suggestion #1",
              "postbackData": "suggestion_1"
            }
          },
          {
            "action": {
              "text": "Suggestion #2",
              "postbackData": "suggestion_2",
              "openUrlAction": {
                "url": "https://www.google.com"
              }
            }
          },
          {
            "action": {
              "text": "Suggestion #3",
              "postbackData": "suggestion_3",
              "dialAction": {
                "phoneNumber": "+13178723000"
              }
            }
          }
        ]
      }
    }
  }
}

```

Carousel:

```

{
  "fallback": "Card #1\nDescription for #1\n\nCard #2\nDescription for #2\n\n
Reply with \"Card #1\" or \"Card #2\"",
  "richCard": {
    "carouselCard": {

```

```

"cardWidth": "MEDIUM",
"cardContents": [
  {
    "title": "Card #1",
    "description": "The description for card #1",
    "suggestions": [
      {
        "reply": {
          "text": "Card #1",
          "postbackData": "card_1"
        }
      }
    ],
    "media": {
      "height": "MEDIUM",
      "contentInfo": {
        "fileUrl": "https://storage.googleapis.com/kitchen-sink-sample-images/cute-dog.jpg",
        "forceRefresh": false
      }
    }
  },
  {
    "title": "Card #2",
    "description": "The description for card #2",
    "suggestions": [
      {
        "reply": {
          "text": "Card #2",
          "postbackData": "card_2"
        }
      }
    ],
    "media": {
      "height": "MEDIUM",
      "contentInfo": {
        "fileUrl": "https://storage.googleapis.com/kitchen-sink-sample-images/elephant.jpg",
        "forceRefresh": false
      }
    }
  }
]
}
}
}

```

## LINE

Flex Message:

```

{
  "type": "flex",
  "altText": "flex message",
  "contents": {
    "type": "bubble",
    "hero": {
      "type": "image",
      "url": "https://scdn.line-apps.com/n/channel_devcenter/img/fx/01_1_cafe.png",
      "size": "full",
      "aspectRatio": "20:13",
      "aspectMode": "cover",
      "action": {

```

```
    "type": "uri",
    "uri": "http://linecorp.com/"
  }
},
"body": {
  "type": "box",
  "layout": "vertical",
  "contents": [
    {
      "type": "text",
      "text": "Brown Cafe",
      "weight": "bold",
      "size": "xl"
    },
    {
      "type": "box",
      "layout": "baseline",
      "margin": "md",
      "contents": [
        {
          "type": "icon",
          "size": "sm",
          "url": "https://scdn.line-apps.com/n/channel_devcenter/img/fx/review_gold_star_28.png"
        },
        {
          "type": "icon",
          "size": "sm",
          "url": "https://scdn.line-apps.com/n/channel_devcenter/img/fx/review_gold_star_28.png"
        },
        {
          "type": "icon",
          "size": "sm",
          "url": "https://scdn.line-apps.com/n/channel_devcenter/img/fx/review_gold_star_28.png"
        },
        {
          "type": "icon",
          "size": "sm",
          "url": "https://scdn.line-apps.com/n/channel_devcenter/img/fx/review_gold_star_28.png"
        },
        {
          "type": "text",
          "text": "4.0",
          "size": "sm",
          "color": "#999999",
          "margin": "md",
          "flex": 0
        }
      ]
    },
    {
      "type": "box",
      "layout": "vertical",
      "margin": "lg",
      "spacing": "sm",
      "contents": [
```

```

    {
      "type": "box",
      "layout": "baseline",
      "spacing": "sm",
      "contents": [
        {
          "type": "text",
          "text": "Place",
          "color": "#aaaaaa",
          "size": "sm",
          "flex": 1
        },
        {
          "type": "text",
          "text": "Miraina Tower, 4-1-6 Shinjuku, Tokyo",
          "wrap": true,
          "color": "#666666",
          "size": "sm",
          "flex": 5
        }
      ]
    },
    {
      "type": "box",
      "layout": "baseline",
      "spacing": "sm",
      "contents": [
        {
          "type": "text",
          "text": "Time",
          "color": "#aaaaaa",
          "size": "sm",
          "flex": 1
        },
        {
          "type": "text",
          "text": "10:00 - 23:00",
          "wrap": true,
          "color": "#666666",
          "size": "sm",
          "flex": 5
        }
      ]
    }
  ]
},
"footer": {
  "type": "box",
  "layout": "vertical",
  "spacing": "sm",
  "contents": [
    {
      "type": "button",
      "style": "link",
      "height": "sm",
      "action": {
        "type": "uri",
        "label": "CALL",

```

```

        "uri": "https://linecorp.com"
      }
    },
    {
      "type": "button",
      "style": "link",
      "height": "sm",
      "action": {
        "type": "uri",
        "label": "WEB",
        "uri": "https://linecorp.com"
      }
    }
  ],
  "flex": 0
}
}
}

```

## Microsoft Teams

Card:

```

{
  "$schema": "https://adaptivecards.io/schemas/adaptive-card.json",
  "type": "AdaptiveCard",
  "version": "1.0",
  "body": [
    {
      "type": "Container",
      "items": [
        {
          "type": "TextBlock",
          "text": "Publish Adaptive Card schema",
          "weight": "bolder",
          "size": "medium"
        },
        {
          "type": "ColumnSet",
          "columns": [
            {
              "type": "Column",
              "width": "auto",
              "items": [
                {
                  "type": "Image",
                  "url": "
https://pbs.twimg.com/profile_images/3647943215/d7f12830b3c17a5a9e4afcc370e3a37e_400x400.jpeg",
                  "size": "small",
                  "style": "person"
                }
              ]
            },
            {
              "type": "Column",
              "width": "stretch",
              "items": [
                {
                  "type": "TextBlock",
                  "text": "Matt Hiding",
                  "weight": "bolder",

```

```

        "wrap": true
      },
      {
        "type": "TextBlock",
        "spacing": "none",
        "text": "Created {{DATE(2017-02-14T06:08:39Z, SHORT)}}",
        "isSubtle": true,
        "wrap": true
      }
    ]
  }
]
},
{
  "type": "Container",
  "items": [
    {
      "type": "TextBlock",
      "text": "Now that we have defined the main rules and features of the format, ↵
we need to produce a schema and publish it to GitHub. The schema will be the starting ↵
point of our reference documentation.",
      "wrap": true
    },
    {
      "type": "FactSet",
      "facts": [
        {
          "title": "Board:",
          "value": "Adaptive Card"
        },
        {
          "title": "List:",
          "value": "Backlog"
        },
        {
          "title": "Assigned to:",
          "value": "Matt Hiding"
        },
        {
          "title": "Due date:",
          "value": "Not set"
        }
      ]
    }
  ]
}
],
"actions": [
  {
    "type": "Action.ShowCard",
    "title": "Comment",
    "card": {
      "type": "AdaptiveCard",
      "body": [
        {
          "type": "Input.Text",
          "id": "comment",
          "isMultiline": true,

```

```

        "placeholder": "Enter your comment"
      }
    ],
    "actions": [
      {
        "type": "Action.Submit",
        "title": "OK"
      }
    ]
  }
},
{
  "type": "Action.OpenUrl",
  "title": "View",
  "url": "https://adaptivecards.io"
}
]
}

```

## Slack

Approval:

```

{
  "blocks": [
    {
      "type": "section",
      "text": {
        "type": "mrkdown",
        "text": "You have a new request:\n*<google.com|Fred Enriquez - Time Off request>*"
      }
    },
    {
      "type": "section",
      "text": {
        "type": "mrkdown",
        "text": "*Type:*\nPaid time off\n*When:*\nAug 10-Aug 13\n ⚡\n*Hours:* 16.0 (2 days)\n*Remaining balance:* 32.0 hours (4 days)\n ⚡\n*Comments:* \"Family in town, going camping!\""
      },
      "accessory": {
        "type": "image",
        "image_url": "https://api.slack.com/img/blocks/bkb_template_images/approvalsNewDevice.png",
        "alt_text": "computer thumbnail"
      }
    },
    {
      "type": "actions",
      "elements": [
        {
          "type": "button",
          "text": {
            "type": "plain_text",
            "emoji": true,
            "text": "Approve"
          },
          "style": "primary",
          "value": "click_me_approve"
        },
        {
          "type": "button",

```



```

    "text": {
      "type": "plain_text",
      "emoji": true,
      "text": "Deny"
    },
    "style": "danger",
    "value": "click_me_deny"
  }
]
}
]
}

```

Notification:

```

{
  "blocks": [
    {
      "type": "section",
      "text": {
        "type": "plain_text",
        "emoji": true,
        "text": "Looks like you have a scheduling conflict with this event:"
      }
    },
    {
      "type": "divider"
    },
    {
      "type": "section",
      "text": {
        "type": "mrkdwn",
        "text": "*<fakeLink.toUserProfiles.com|Iris / Zelda 1-1>*\n ↵\n\nTuesday, January 21 4:00-4:30pm\nBuilding 2 - Havarti Cheese (3)\n2 guests"
      },
      "accessory": {
        "type": "image",
        "image_url": "https://api.slack.com/img/blocks/bkb_template_images/notifications.png",
        "alt_text": "calendar thumbnail"
      }
    },
    {
      "type": "context",
      "elements": [
        {
          "type": "image",
          "image_url": "↵\n\nhttps://api.slack.com/img/blocks/bkb_template_images/notificationsWarningIcon.png",
          "alt_text": "notifications warning icon"
        },
        {
          "type": "mrkdwn",
          "text": "*Conflicts with Team Huddle: 4:15-4:30pm*"
        }
      ]
    },
    {
      "type": "divider"
    },
    {
      "type": "section",

```

```
"text": {
  "type": "mrkdwn",
  "text": "*Propose a new time:*"
}
},
{
  "type": "section",
  "text": {
    "type": "mrkdwn",
    "text": "*Today - 4:30-5pm*\nEveryone is available: @iris, @zelda"
  },
  "accessory": {
    "type": "button",
    "text": {
      "type": "plain_text",
      "emoji": true,
      "text": "Choose"
    },
    "value": "click_me_today_1630"
  }
},
{
  "type": "section",
  "text": {
    "type": "mrkdwn",
    "text": "*Tomorrow - 4-4:30pm*\nEveryone is available: @iris, @zelda"
  },
  "accessory": {
    "type": "button",
    "text": {
      "type": "plain_text",
      "emoji": true,
      "text": "Choose"
    },
    "value": "click_me_tomorrow_1600"
  }
},
{
  "type": "section",
  "text": {
    "type": "mrkdwn",
    "text": "*Tomorrow - 6-6:30pm*\nSome people aren't available: @iris, ~@zelda~"
  },
  "accessory": {
    "type": "button",
    "text": {
      "type": "plain_text",
      "emoji": true,
      "text": "Choose"
    },
    "value": "click_me_tomorrow_1800"
  }
},
{
  "type": "section",
  "text": {
    "type": "mrkdwn",
    "text": "*<fakeLink.ToMoreTimes.com|Show more times>*"
  }
}
}
```

```
]
}
```

## Viber

Sticker (Bowling Cat):

```
{
  "type": "sticker",
  "stickerId": "bfe458bf64fa8da4d4eb"
}
```

Carousel:

```
{
  "type": "rich_media",
  "rich_media": {
    "Type": "rich_media",
    "ButtonsGroupColumns": 6,
    "ButtonsGroupRows": 7,
    "BgColor": "#FFFFFF",
    "Buttons": [
      {
        "Columns": 6,
        "Rows": 3,
        "ActionType": "open-url",
        "ActionBody": "https://www.google.com",
        "Image": "
https://image.made-in-china.com/44f3j00oHRakwIAaLzT/
Custom-Logo-Glowing-Stereo-Headphones-Computer-Gaming-Headset-with-Microphone.jpg"
      },
      {
        "Columns": 6,
        "Rows": 2,
        "Text": "<font color=#323232><b>Headphones with Microphone, ↵
On-ear Wired earphones</b></font>
<font color=#777777><br>Sound Intone </font><font color=#6fc133>$17.99</font>",
        "ActionType": "open-url",
        "ActionBody": "https://www.google.com",
        "TextSize": "medium",
        "TextVAlign": "middle",
        "TextHAlign": "left"
      },
      {
        "Columns": 6,
        "Rows": 1,
        "ActionType": "reply",
        "ActionBody": "https://www.google.com",
        "Text": "<font color=#ffffff>Buy</font>",
        "TextSize": "large",
        "TextVAlign": "middle",
        "TextHAlign": "middle",
        "Image": "
https://img.favpng.com/25/23/15/green-area-angle-png-favpng-zEacBKmPkyuj2T0RzFWhaeWHx.jpg"
      },
      {
        "Columns": 6,
        "Rows": 1,
        "ActionType": "reply",
        "ActionBody": "https://www.google.com",
        "Text": "<font color=#8367db>MORE DETAILS</font>",

```

```

        "TextSize": "small",
        "TextVAlign": "middle",
        "TextHAlign": "middle"
    },
    {
        "Columns": 6,
        "Rows": 3,
        "ActionType": "open-url",
        "ActionBody": "https://www.google.com",
        "Image": "\u2197"
    }
],
"Image": "https://shoppingu.today/wp-content/uploads/2018/06/I-am-Thinking-T-shirt.jpg"
},
{
    "Columns": 6,
    "Rows": 2,
    "Text": "<font color=#323232><b>Hanes Men's Humor Graphic T-Shirt</b></font>\u2197"
    <font color=#777777><br>Hanes</font><font color=#6fc133>$10.99</font>",
    "ActionType": "open-url",
    "ActionBody": "https://www.google.com",
    "TextSize": "medium",
    "TextVAlign": "middle",
    "TextHAlign": "left"
},
{
    "Columns": 6,
    "Rows": 1,
    "ActionType": "reply",
    "ActionBody": "https://www.google.com",
    "Text": "<font color=#ffffff>Buy</font>",
    "TextSize": "large",
    "TextVAlign": "middle",
    "TextHAlign": "middle",
    "Image": "\u2197"
}
],
"Image": "https://img.favpng.com/25/23/15/green-area-angle-png-favpng-zEacBKmPkyuj2T0RzFWhaeWHx.jpg"
},
{
    "Columns": 6,
    "Rows": 1,
    "ActionType": "reply",
    "ActionBody": "https://www.google.com",
    "Text": "<font color=#8367db>MORE DETAILS</font>",
    "TextSize": "small",
    "TextVAlign": "middle",
    "TextHAlign": "middle"
}
]
}
}
}

```

## Zalo

List:

```

{
  "attachment": {
    "type": "template",
    "payload": {
      "template_type": "list",
      "elements": [
        {
          "title": "Official Account API",

```

```
    "subtitle": "Zalo API",
    "image_url": "https://stc-developers.zdn.vn/images/bg_1.jpg",
    "default_action": {
      "type": "oa.open.url",
      "url": "https://developers.zalo.me/docs/api/official-account-api-147"
    }
  },
  {
    "title": "Article API",
    "image_url": "https://stc-zaloprofile.zdn.vn/pc/v1/images/zalo_sharelogo.png",
    "default_action": {
      "type": "oa.open.url",
      "url": "https://developers.zalo.me/docs/api/article-api-151"
    }
  },
  {
    "title": "Social API",
    "image_url": "https://stc-zaloprofile.zdn.vn/pc/v1/images/zalo_sharelogo.png",
    "default_action": {
      "type": "oa.open.url",
      "url": "https://developers.zalo.me/docs/api/social-api-4"
    }
  },
  {
    "title": "Shop API",
    "image_url": "https://stc-zaloprofile.zdn.vn/pc/v1/images/zalo_sharelogo.png",
    "default_action": {
      "type": "oa.open.url",
      "url": "https://developers.zalo.me/docs/api/shop-api-124"
    }
  }
]
}
```