



# Bots

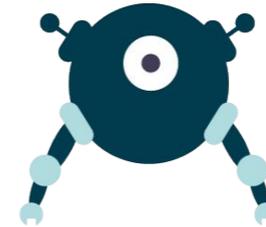
uContact net2phone | Training.



## What is a Bot?

A Bot is a computer program that automatically performs repetitive tasks that would otherwise be extremely tedious, difficult or impossible for a human being to do.

Bots allow the execution of workflows using a JavaScript motor that enables them to perform almost any action if they have the necessary knowledge-base. They can do anything from simple answers to queries to a database, transactions, Web Service executions, and more.



<ChatBot



**Samantha**

Hi, I need some information

**Integra Chatbot**

Welcome to our automatic attention system.  
Who do you want to talk to?

1. Support Team
2. Sales Team
3. Customer Success



**Samantha**

3 Customer Success please

**Channels supported**



Bots



**Webchat**



**WhatsApp**



**Facebook  
Messenger**



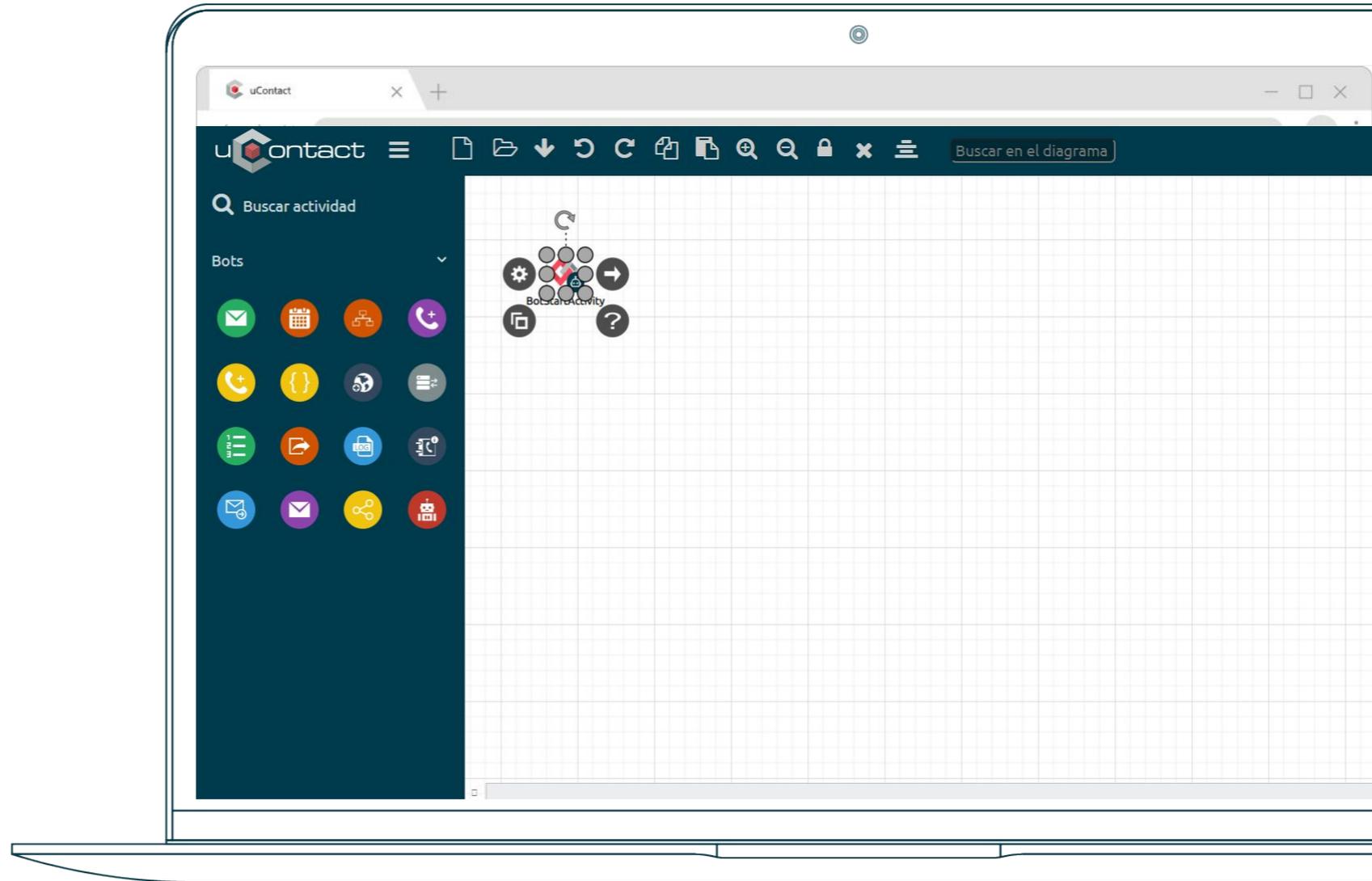
**SMS**



**Mail**

Log into uContact with a **Supervisor account**, which has permission to use its **Workflow Designer**.

1. Once inside the platform, access the Workflow Designer. There, you will find the option to create a specific workflow for Bots.
1. Choose and select the 'BotStartActivity' icon and, with a simple Drag-and-Drop system, drag all the activities you will ask the Bot to perform. This way, you are building its workflow.



The tasks and **activities** you ask the Bot to do are the **nodes that make up the workflow and determine its behaviour**. In order to create and develop your Bot, you will need to mix and combine different activities and build your workflow according to the actions you want it to perform.





## BotStartActivity:

This is the first activity on your Bot's workflow and, as its namesake gives away, it orders the Bot to begin its activity, Therefore, this activity must be present in every workflow you create if you want the Bot to perform correctly.

### Fields:

**name**

Name of the Bot.

**botTimeout**

Time (in minutes) that the Bot will have to wait without receiving any response before finishing its activity..



## SendMessage:

This activity allows the Bot to send clients **three types of messages**:

### Simple messages

To do this, we simply have to type the message you wish the Bot to send in the 'Message' field.

### Message with options

After introducing itself to the client, the Bot might want to give him different options and then make decisions based on his answer. In order to do so, you'll need to state the options in the 'Message' field of the activity; defining the 'Option Number' and separating it with an 'Enter'.

### Message with results

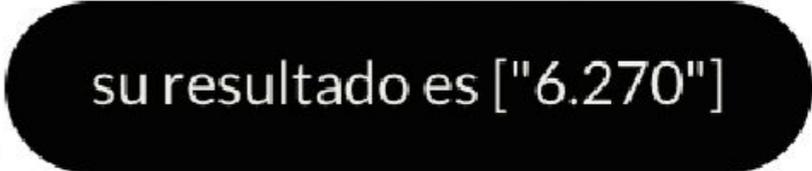
Other Bot Activities enable it to create and storage data in variables, which then can be shown and forwarded to clients as results to their questions.

### For example:

If your workflow consults a certain database in search for a client's number, you can store this information with a variable and then use a 'SendMessage' to send the client a message that says:

**"Your client number is `${Variable}`"**

The client will therefore receive a message with the result of the query where it says '`${variable}`'.



su resultado es ["6.270"]



### IsOnTime:

This activity allows the Bot to evaluate if the time introduced for the activity is within the set time period. If it is, the Bot will continue to perform the activities in the green path of the workflow; otherwise, it will have to continue through the red one.

#### Example:

The client sends a message to the Bot and selects the option 'Talk to an agent'. After receiving the client's answer, the Bot can decide if it wants to transfer him to a campaign or another, depending on the set time period.

#### The format of the schedule must be:

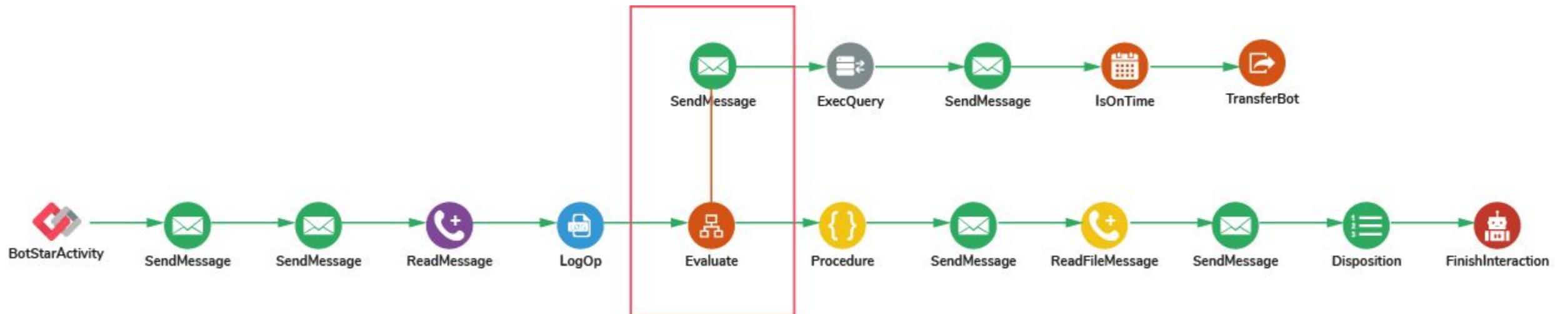
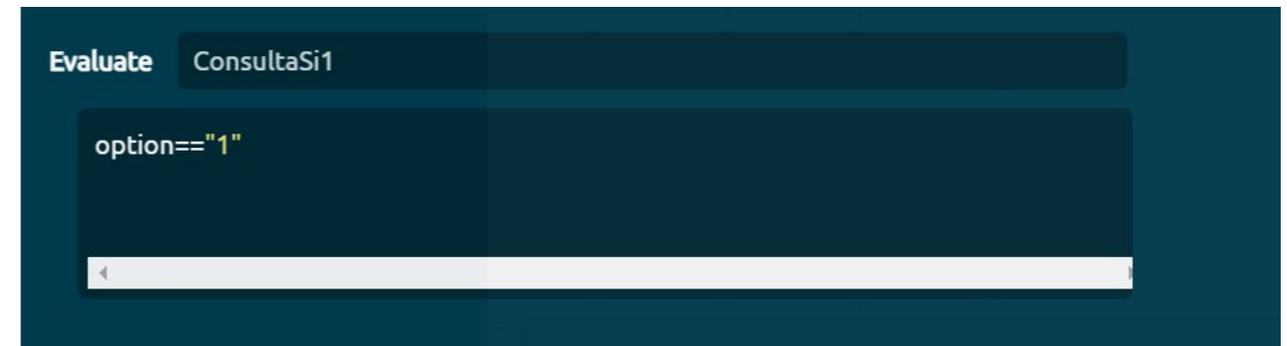
mon,tue,wed,thu,fri,sat,sun;09:00-18:00

#### And can concatenate multiple schedules with &:

mon,tue,wed,thu,fri;09:00-18:00&sat,sun;09:00-13:00

## Evaluate:

Allows the Bot to evaluate a client's answer and, based on it, decide whether to follow one path or the other. This type of activities can be chained one after the other in those cases where multiple options result in different actions. If the variable is fulfilled, it will go through the green path that emerges from the 'Evaluate' nodule and, if not, it will go through the red one.





## ReadMessage:

This activity enables the Bot to read the answer sent by the client and then evaluate it or save it on a variable.

ReadMessage	Description
variable	0



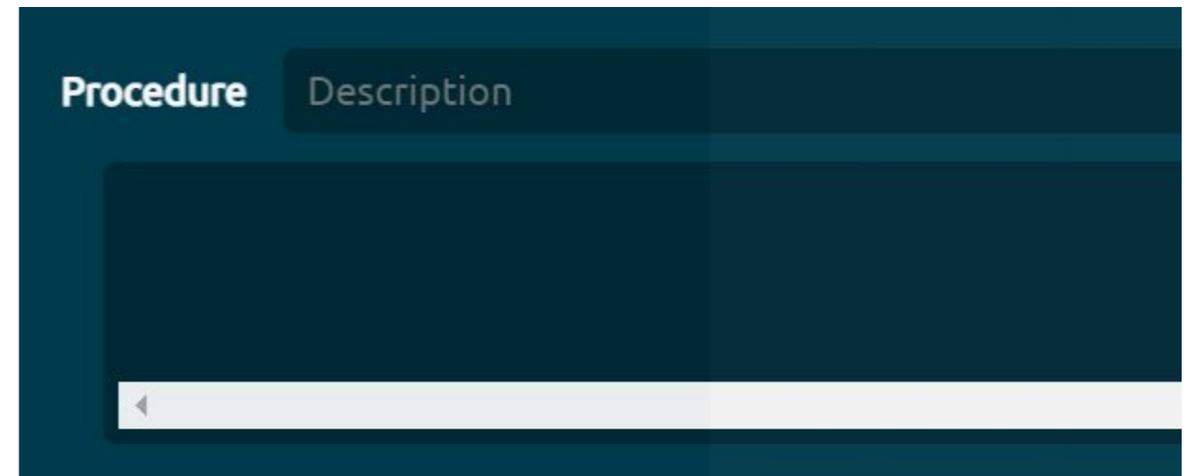
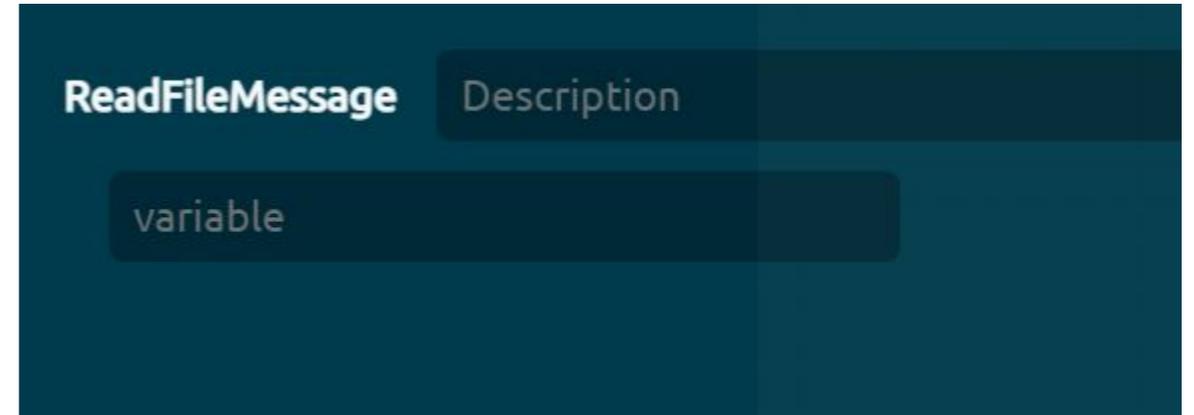
## ReadFileMessage:

This activity is similar to the previous one, with the only difference that it is used to save a file and successfully forward it to the client.



## Procedure:

The 'Procedure' activity enable the Bot to execute a JavaScript code to parse an answer that comes from a Web Service, increase a variable's value, create functions for future use, and many other things.

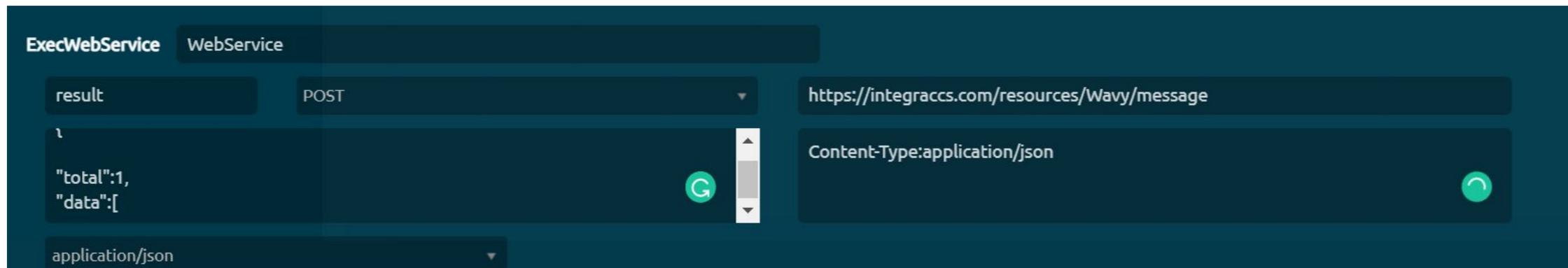




## ExecWebService:

As its namesake gives away, it allows the Bot to execute a Web Service. In order to do so, the following fields are required:

variable	method	URL	body	headers	content-type
Name of the variable where the Web Service's answer will be configured.	The Web Service's method.	Web Service's URL.	The content of the request.	They are written in lowercase and separated with an 'Enter'.	Header type that comes with preset options to facilitate the content upload.





## ExecQuery:

This activity allows the Bot to make queries to a database. In order to do so, it takes into account the following fields:

database	variable	script
Name of the database you wish to consult.	Name of the variable where the Web Service's answer will be configured.	

ExecQuery ConsultaVersion

data

```
select value from ccdata.configuraciones where config = 'version'
```



## Disposition:

This activity allows the Bot to typify all its interactions, making it possible to give follow-ups to clients. All the dispositions are inserted directly into the BD, therefore, they do not need to be created in the system. Also, even if they are created in uContact, since they belong to another channel, they will not execute actions such as 'respool' or 'reschedule'.

The screenshot shows a configuration window for a 'Disposition' activity. The title bar reads 'Disposition' and the instance name is 'Dispositionlf4'. The configuration includes a 'Select' dropdown menu set to 'Number' and a value field containing '4'. Below this, a text box displays the message 'Este usuario seleccionó 4' with a green circular refresh icon to its right. On the far right, there is a text field containing the word 'extra'.

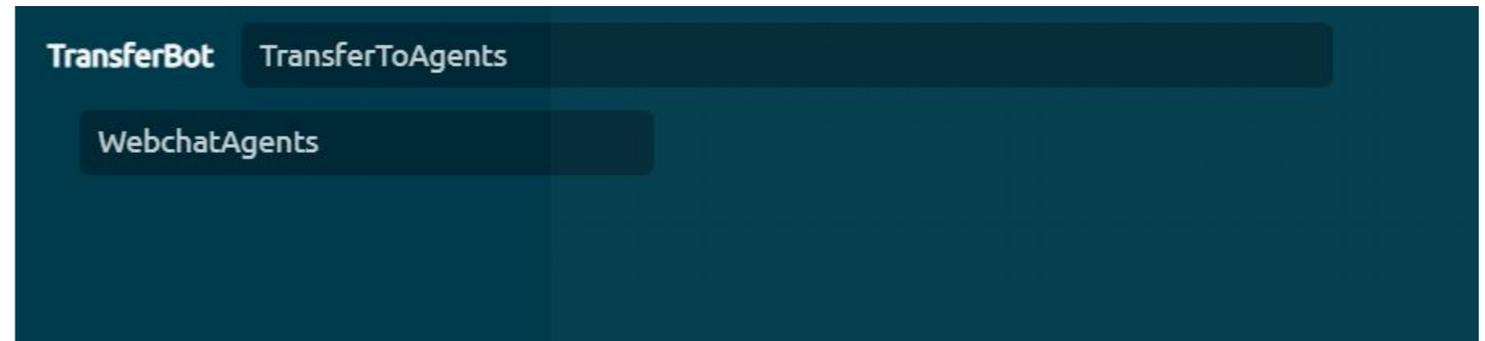


## TransferBot:

With this activity, it is possible to transfer a Bot's interaction to a campaign that uses the same channel (WebChat, WhatsApp, SMS, and more). To do it correctly, you should indicate the name of the campaign you wish to transfer the interaction to.

Should you use this activity, it is recommended that you do not use the same Bot for two campaigns with different channels, since the Bot will not be able to transfer its interactions to another channel.

**Note:** You cannot include a 'FinishInteraction' after applying a 'TransferBot' activity.





### LogOp:

Allows the Bot to print a result or an answer in the 'IntegraServer' log, with SEVERE mode.

#### Format

`${log}`

```
LogOp CreateOptionValue
```

```
  ${result}
```



## FinishInteraction:

This activity indicates the Bot that his job has been completed and finished its interactions with the client. Therefore, it should be the last node on its workflow.

After a 'FinishInteraction', the client will be given the option to stay or abandon the conversation, but will be unable to continue writing or calling inside the same chat..

FinishInteraction	Description

**Bot Function.**

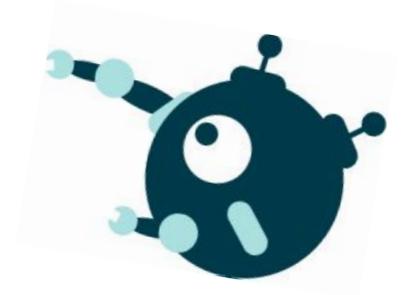


Bot Functions are a type of Bots that, as its name indicates, are used as functions. Their purpose is to be able to modularize certain sections of the Bots, in order to facilitate their development, testing and maintenance. To create a 'BotFunction', you must add the 'StartBotFunction' activity, that is in the same menu as the Bots' one.

Unlike the 'BotStartActivity', this type of Bot only has the 'name' field. In other words, a Bot Function's initial activity does not have a 'bot\_timeout' or 'timeout\_message' field, which means that the timeout that will be taken into account is the one that calls the Bot Function into action.



### BotFunctionStartActivity



**Note:** Bot Functions do not use up Bots' licences and are uploaded into the server with 'botsFunction' in their body text.

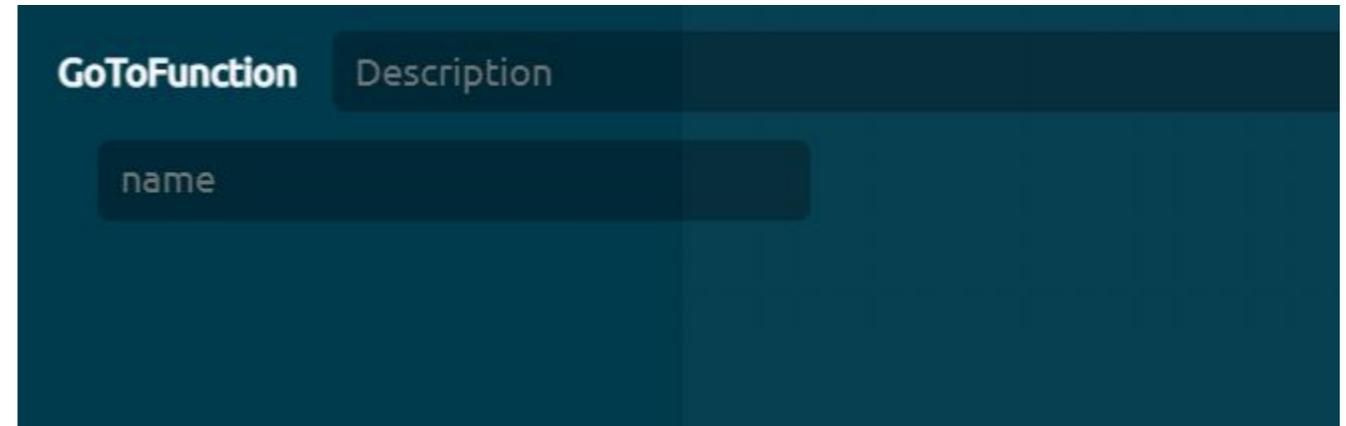
**Go-To-Function.**



### GoToFunction:

The 'Go-To-Function' is an activity that allows a Bot and a Bot Function to be linked. For instance, if you have a Bot named 'B' and a Bot Function called 'BF', and 'B' has a 'Go-To-Function' activity associated with the name-field 'BF'; when this activity is executed, 'B' will pause its work and 'BF' will start. Then, when the workflow is done, 'B' will resume its work.

This way, you can **link Bots and Bot Functions in an initial Bot workflow.**



#### name

Field that contains the Bot's or Bot Function's name to which you wish to give up control once the 'Go-To-Function' is executed.

**Global variables of a  
BOT.**

Certain information about clients can be obtained from different sources and interaction, which can be used for future instances.

Therefore, Bots have different **global** (equal for all the channels) and **particular** (different for each channel) variables associated.

The global ones are:

For all the channels:

- **CHANNEL**
- **CLIENT\_ID**

For each channel:

Webchat

- **PHONE\_NUMBER**
- **CLIENT\_NAME**

SMS - WhatsApp

- **SMS\_PROVIDER\_TYEPE**

Messenger:

- **CLIENT\_NAME**
- **PAGE\_NAME**

Mail:

- **EMAIL\_SUBJECT**
- **EMAIL\_BODY**

## Global variables of a Bot | To be considered

Before moving forward to explaining how to implement a Bot, we should revise the following aspects, which are of fundamental importance:



BotStarActivity

### BotStartActivity

A Bot's workflow must always start with the 'BotStartActivity' node.



FinishInteraction

### FinishInteraction

Always end the Bot's workflow with the 'FinishInteraction' node.



TransferBot

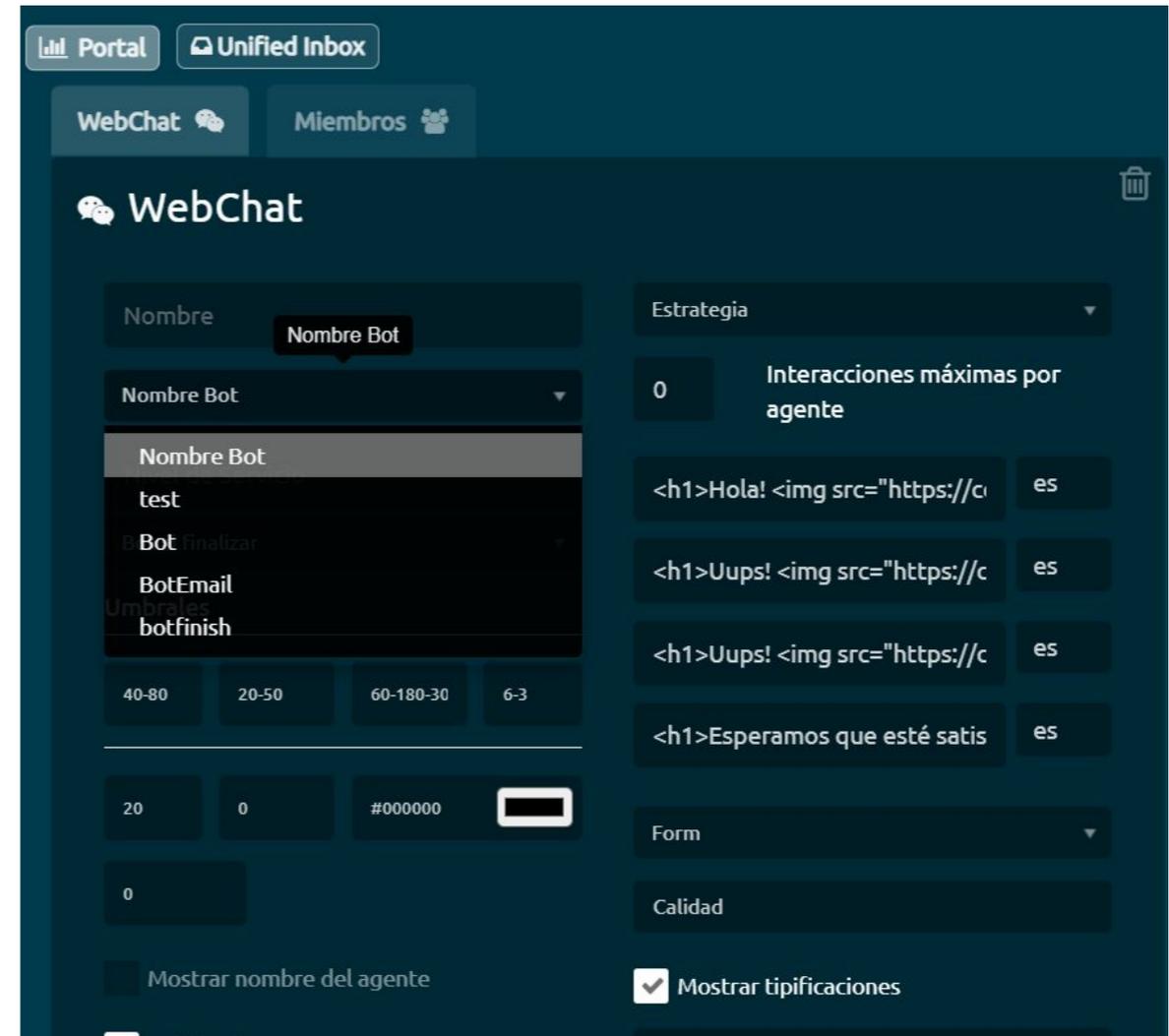
### TransferBot

After a 'TransferBot' activity you must not include the 'FinishInteraction' node.

**Bot Administration.**

To create or start configuring a Bot, you must access the **'Campaign Setup'** section of the channel in which you wish to implement it (SMS, WebChat, and others) and, inside the deployed menu, select the option **'Bot Name'**. When deploying it, all the created and available to select Bot workflows will appear.

Once you selected the Bot you wish to execute for the campaign, just **press 'Save'** and the Bot will be active for that campaign.





There are two ways to use a Bot:

- 1. Assign the Bot to a campaign:** If it has a Bot associated, all the interactions that enter that campaign will be attended by the Bot, even if it already has agents assigned.
- 1. With the activity 'BotWhenFinish':** This works like a campaign with agents, but, when the agent finished the interaction, it will not really end, it will be transferred to a Bot and will finalize when its workflow comes to an end. In general, this way is used to send satisfaction surveys at the end of an interaction.

All the **interactions that are active or in a waiting list** when making any change in the campaign **will be affected.**

Campaigns that have Bots do not need to have active agents, since the Bots will take the interactions and distribute them to other, more specific campaigns..

The **interactions that are generated after adding a Bot** to the campaign **will start using it.**

Chats that use Bots are recorded and can be consulted in the 'Recordings' section of the channel in question.

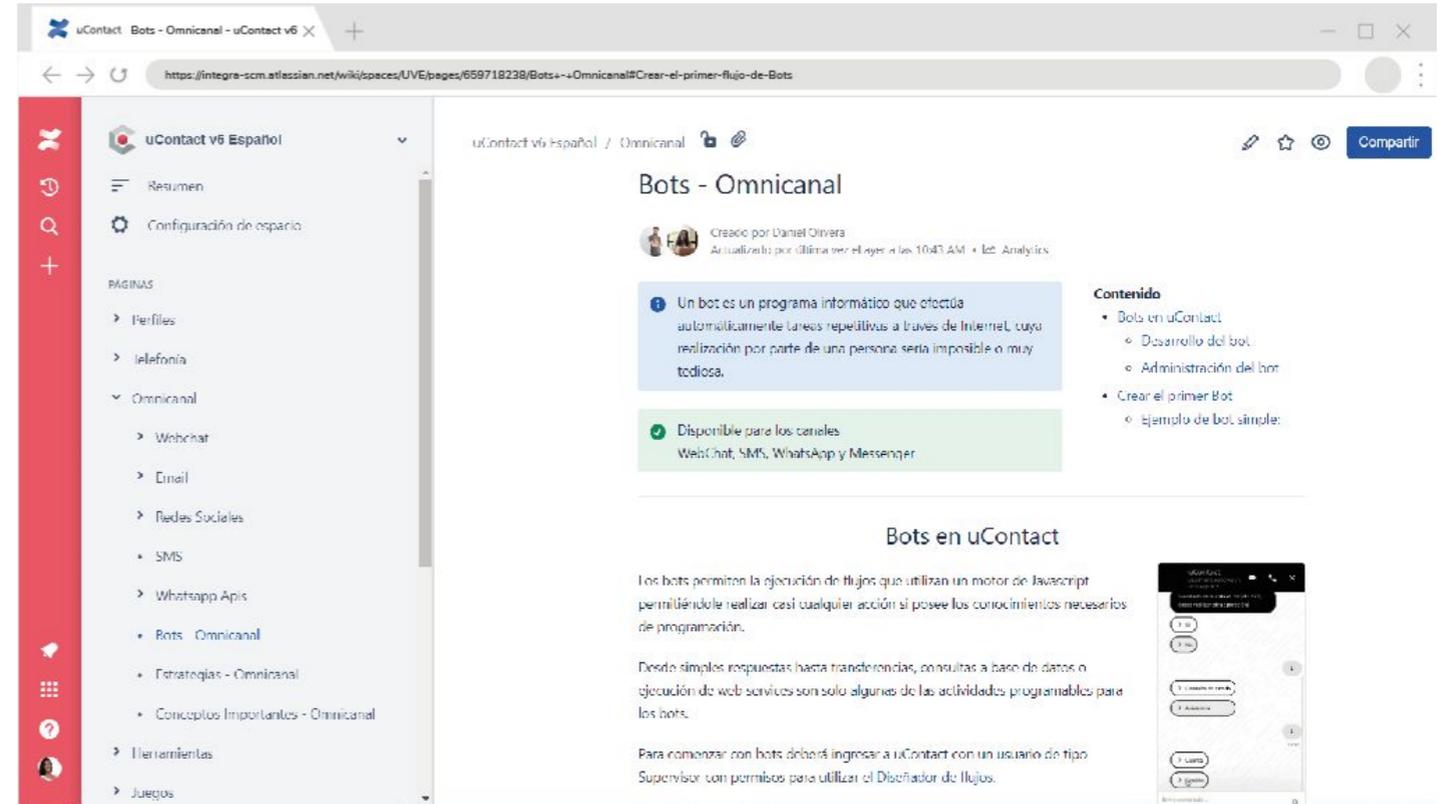
Bots imitate or replace the function of an agent, which allows them to create reports from data obtained in its interactions.

In each channel's 'Messages' section, the 'agent' parameter is the Bot's name.

For more information and a step-by-step guide about how to setup a Bot in a campaign, click on the following link:

[Bot Designer: A step-by-step guide](#)

Also, you can request access to a .json document with an example of an already existing Bot workflow by getting in touch with us at [info@integraccs.com](mailto:info@integraccs.com).





**Thanks!**

