

getCallerInformation

This API works in tandem with an IVR object in the jtel system, which can be used to search for records in SalesForce when calls are received.

The results of the search performed are available as variables, and can be used to influence the routing in the jtel system, for example to route to a particular agent, or to a particular ACD group.

To make use of this API, a REST endpoint must be provided in SalesForce for the jtel System to access. The following pages describe an example implementation.

Request

GET

Parameters

The following parameters are provided in the REST request:

Parameter Name	From jtel Release	Description
CallerID		The number of the caller in E.164 (fully qualified without +) format.
ServiceNumber		The service number called (jtel field ServiceNumbers.Name).
ServiceName		The service name called (jtel field ServiceNumbers.Name2).
Function	3.25 / 2021-07-02 UPWARDS	The function to execute. This parameter can be used to differentiate the required action and could be used for many business cases. For example, to search for a ticket. Or event to update SalesForce and create an object.
UserData	3.25 / 2021-07-02 UPWARDS	A user definable parameter which can contain any information.
UserData2	3.25 / 2021-07-02 UPWARDS	A user definable parameter which can contain any information.
UserData3	3.25 / 2021-07-02 UPWARDS	A user definable parameter which can contain any information.

http Status Codes

The following status codes should be used:

Status Code	From jtel Release	Meaning
200 – OK		One record was found and returned.
300 – Multiple Choices	3.29 / 2022-08-08 AND EARLIER	Multiple records were found. (No records should be returned in this case).

300 – Multiple Choices	3.30 / 2022-08-08 AND LATER	<p>Multiple records were found. The Apex class can either:</p> <ul style="list-style-type: none"> ▪ return no records ▪ return one of the found records In this case, if the record is to be used for a screen-pop (by populating the UserData field) some information should be provided to the agent that more than one record in fact exist, for example in the additional information field. ▪ return all of the found records in an array. In the last case, the logic in the routing application will be correspondingly more difficult, but the routing application will be given access to all returned records in variables with additional indexing. For example <code>\$\$SalesForce.0.ID ... \$\$SalesForce.1.ID</code> and so on.
404 – Not found		No record was found.

Return Structure

The return structure must be JSON, but otherwise any fields and any content can be returned. The jtel System will parse the data into variables for the jtel IVR GUI, of the form **`$$SalesForce.FieldName`**

If structures are nested, then the variable will be named accordingly: **`$$SalesForce.Entity.FieldName`**

Examining the call log will show you the exact field names returned.