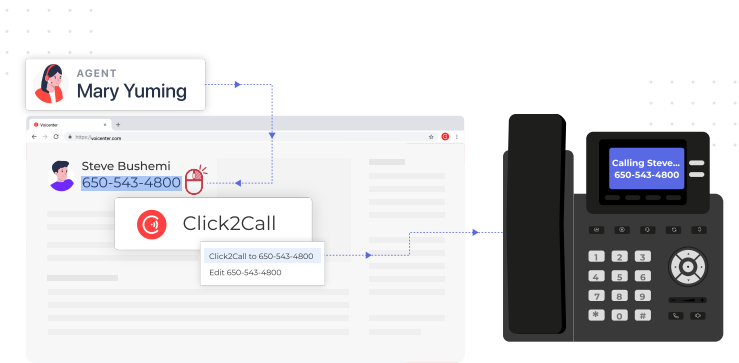


# Click To Call API By Voicenter

Click2Call enables executing direct phone calls, with just the click of a button. It's super easy to implement and integrate, using our ready-to-use code. This great and versatile API integration can really make a difference, wherever you assimilate it



Initiating phone-calls from any available representative or salesperson -automatically!(Great performance booster, fully personalize to any duration).

Trigger and initiate an automatic call-back to any 'Contact us' lead or customer, the moment they click the 'Apply/Send' button. Imagine their WOW, and your conversion rates.

Integrate click2call to your CRM, will improve your performance and make sure you fully-deliver your SLA ([http://en.wikipedia.org/wiki/Service-level\\_agreement](http://en.wikipedia.org/wiki/Service-level_agreement)) as endorsed.

## Click To Call API Works As Follows:

The API for each call you initiate activates **2 "legs"**:

1. **"Leg 1"** is the first call the agent receives to his extension after initiating a call. This is set by the "Phone" parameter where you input the ID of the agent's extension.
2. **"Leg 2"** is the call to the chosen destination (following an answer to the leg 1 call), this is set by configuring the client's phone number in the target parameter.

## General Click2Call API description:

### API URI

```
1 | https://api.voicenter.com/ForwardDialer/click2call.aspx
```

## Capabilities of the API

This API allows you to initiate any of the following 3 types of calls:

1. Simple outgoing call.
2. Callback and click to IVR.
3. Terminate calls.

## Acceptable request types

1. POST-JSON
2. GET

## Possible Response formats

To control the format of the response please provide your desired format in the request:

- 1. XML (default)
- 2. POST-JSON

### Request Example GET Format

```
1 https://api.voicenter.com/ForwardDialer/click2call.aspx?phone=XXXXXXXX&target=057XXXXXX&code=XXXXXXXXXX&action=c
```

### Request Example POST-JSON Format

```
1 {  
2   "code": "XXXXXXXXXX",  
3   "phone": "APIAPIAPI",  
4   "action": "call",  
5   "target": "0512345678"  
6 }
```

### Call Action Request Parameters

| Parameter       | Description  | Appearance |
|-----------------|--|------------|
| phone           | First destination to dial.<br>Phone number or Voicenter extension.<br>Phone must be in E164 format, but without the plus sign.<br>If the phone is Israeli,format can be 0*****.<br>In case of dialing to an extension behind a SIP Trunk, please set<br>Phone=SIPTRUNK.<br>In case of extension dependency on user login please set:<br>phone=logincode phonelogincode=[login code]. | Mandatory  |
| target          | Second destination to dial.<br>Phone number or Voicenter extension.<br>Phone should be in E164 format, but without the leading plus (+) sign. If<br>the Target is Israeli, format can be 0***** too.   | Mandatory  |
| code            | Caller identity token, attainable from our back office department.   | mandatory  |
| action          | Call (action-call).  | mandatory  |
| record          | Record the call.<br>Acceptable values: true / false.<br>If not provided set to false.  | Optional   |
| phonecallerid   | Caller id of the phone.<br>Not supported by all providers.   | Optional   |
| phonecallername | Caller name of the phone.<br>Not supported by all providers.   | Optional   |
| targetcallerid  | Caller id of the target.<br>Not supported by all providers.  | Optional   |

| Parameter              | Description   | Appearance |
|------------------------|---|------------|
| targetcallername       | Caller name of the target.<br>Not supported by all providers.   | Optional   |
| phonemaxdialtime       | Phone maximum dialtime, in seconds.<br>If not provided set to 60 seconds.   | Optional   |
| targetmaxdialtime      | Target maximum dial time, in seconds.<br>If not provided set to 60 seconds.   | Optional   |
| maxduration            | Maximum call duration in seconds.<br>If not provided set to 7200 seconds.   | Optional   |
| phoneautoanswer        | Only for Voicenter extensions.<br>Expecting true or 1 for phone automatic answer.<br>If not specified set to false.   | Optional   |
| targetautoanswer       | Only for Voicenter extensions.<br>Expecting for true or 1 for target automatic answer<br>If not specified set to false.   | Optional   |
| checkphonedevicestate  | Block call if extension offline.  | Optional   |
| checktargetdevicestate | Block call if extension offline.  | Optional   |
| language               | Specify languageforsystem recordings/prompts/etc.<br>Acceptable format is [he/en/ru/ etc...].   | optional   |
| format                 | Controls the format of the response.<br>Possible values : 1. XML - for XML format<br>2. json - for JSON format<br>The default is XML.   | Optional   |
| var_*                  | Any custom parameter with “var_” prefix.<br>For example: var_customer=2342<br>This parameter used in:<br>1.Voicenter chrome notification extension<br>2.Voicenter desktop notification program<br>3.CDR to CRM API<br>The limit is 10 parameters.<br>https://api.voicenter.co.il/ForwardDialer/click2call.aspx?<br>phone=XXXXXXXXX&target=057XXXXXXXXX&code=XXXXXXXXXXXX&action=call<br>&var_accountID=12345678&var_agentID=1234567   | Optional   |
| ignoredncstatus        | In case you activated Voicenter Do-Not-Call-Me service,<br>a service that checks whether to allow calling Israeli destinations,<br>by adding this field to your click2call requests, you will be allowed to make<br>calls to restricted destinations.<br>Here are the values you can pass:<br>1 - Allow calling restricted destinations in the "phone" field.<br>2 - Allow calling restricted destinations in the "target" field.<br>3 - Allow calling restricted destinations in both "phone" and "target" fields. | Optional   |

## Terminate Action

The Click-To-Call provides you with the option not just initialing calls but also allow the agent to end an active call.

## Table Parameters

| Parameter | Description   | Appearance |
|-----------|---|------------|
| Phone     | Voicenter extension To hang up.   | Mandatory  |
| code      | Caller identity for you account, provided by our Backoffice department. | Mandatory  |
| action    | terminate (action=terminate).   | Mandatory  |

## Click To Call API Response

After you make an api request we provide with you with a response to signal if your request was successful or not. The response will be in XML format (XML-RPC structure) or JSON.

The fields are as follows:

1. **ERRORCODE** (Integer) List:

| Code | Description                               |
|------|---|
| 0    | OK  |
| 1    | Invalid request parameters                |
| 2    | Application error                         |
| 3    | The representative's extension is offline |
| 4    | Extension blocked for Click2Call calls    |

2. **ERRORMESSAGE** (String) - A short sentence describing the issue.
3. **CALLID** - String (32 chars), Unique code of the call, in case of failure is empty.

## XML Response Example

```
1  <methodResponse>
2  <params>
3  <param>
4  <value>
5  <struct>
6  <member>
7  <name>ERRORCODE</name>
8  <value><string>1</string></value>
9  </member>
10 <member>
11 <name>ERRORMESSAGE</name>
12 <value><string>Phone is missing</string></value>
13 </member>
14 <member>
15 <name>CALLID</name>
16 <value><string></string></value>
17 </member>
18 </struct>
19 </value>
20 </param>
21 </params>
22
```



```
</methodResponse>
```

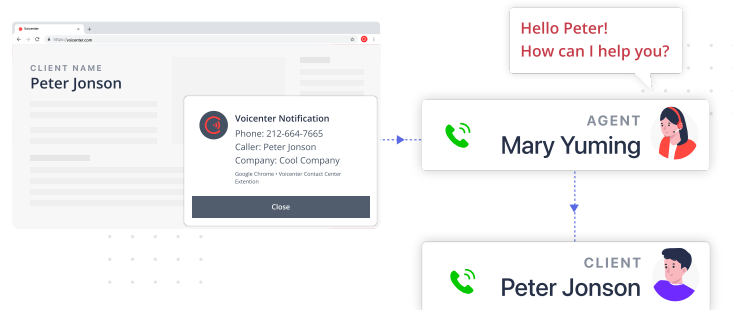
## JSON Response Example

```
1  {
2    "ERRORCODE":0,
3    "ERRORMESSAGE":"OK",
4    "CALLID":"sdfsdfsdfsdfsdfsdfsdfsdfsdf0brp"
5  }
```



# Pop Up Screen API

Use this API to let you pop up a screen with the caller's contact details right out of your business information system (CRM).

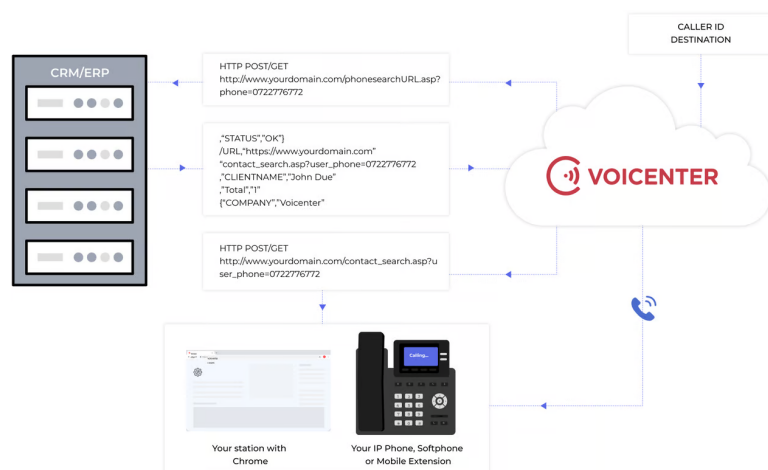


## General Description

If you are working on Chrome browser then you can use our chrome extension and give your agents/representatives the option of popping up the calling customer data from your CRM during incoming calls.

This way your agents will know with which customer is calling even before answering the call.

## How does the Voicenter pop-up screen work?



A caller is calling. The caller Caller-ID is gathered by Voicenter Cloud. Voicenter cloud sends the call data to your Server using simple http request After the client got the call data from Voicenter request, a JSON response is sent back to Voicenter cloud. A notification with the client's response will pop up. Pressing the notification will load a new web chrome browser tab with the provided "URL" link from Step 3.

## Request Formats

1.GET

- 2. POST- Application/X-WWW-Form-Urlencoded
- 3. POST-JSON

## Response Format

- 1. POST-JSON

## Get Request Example

```
1 | https://www.yourdomain.com/contact_search.asp?phone=0722776772&ivrid=202010111sdsd3684752bcb3d&extenU
2 | ser=SIPSIPl&did=0722776773&statusCall=Ringin
```

## POST Request Example

```
1 | phone=0722776772&ivrid=202010111sdsd3684752bcb3d&extenUser=NaDzw50l&did=0722776773&status=Ringin
```

## POST JSON Request Example

For each phase of the call (ringing, answer, hang up) you can receive an api request in JSON format which will contain additional to allow you to better implement your business logic with the popup screen api service.

## Ringin

```
1 | {
2 |   "phone": "972722776772",
3 |   "callerName": "Queue Testing",
4 |   "ivrid": "20220916103546022555eb61e755c08a",
5 |   "extenUser": "KGpK4iWq",
6 |   "did": "0776707528",
7 |   "status": "Ringin",
8 |   "direction": "Incoming",
9 |   "recordFile": "20220916103546022555eb61e755c08a-nikitaapi-972722776772-0776707528.mp3",
10 |   "isMuted": false,
11 |   "callStarted": 1663324548,
12 |   "isAnswered": false,
13 |   "callAnswered": 0,
14 |   "currentCall": {
15 |     "callStarted": 1663324548,
16 |     "calldurationinterval": 1663324548,
17 |     "callAnswered": 0,
18 |     "answered": 0,
19 |     "callername": "Queue Testing",
20 |     "callerphone": "972722776772",
21 |     "outgoingcallername": "",
22 |     "outgoingcallerphone": "",
23 |     "callstatus": "Ringin",
24 |     "customdata": {
25 |     }
26 |   },
27 |   "direction": "Incoming",
28 |   "ivrid": "20220916103546022555eb61e755c08a",
29 |   "recordin": {
30 |     "Filename": "20220916103546022555eb61e755c08a-nikitaapi-972722776772-0776707528.mp3",
31 |     "Options": "",
32 |     "ApproximateURL": "",
33 |     "IsMuted": 0
34 |   }
35 | }
```

```

34     "did": "0776707528",
35     "relatedIvrUniqueIDs": [
36     ]
37     "callType": "Incoming",
38     "originalCallerID": "0722776772",
39     "originalCallerName": "Nikita Karpushin new",
40     "actualDialedNumber": "0776707528",
41     "channel": "SIP/KGpK4iWq-00000007",
42     "channel2": "SIP/ProviderProxy05-00000006",
43     "isSpied": false,
44     "isInternal": false,
45     "ip": "185.138.169.59",
46     "blcServerID": 225,
47     "extenUser": "KGpK4iWq",
48     "number": 89312
49 }
50 }

```

## Answer Phase

```

1  {
2    "phone": "972722776772",
3    "callerName": "Queue Testing",
4    "ivrid": "20220916103546022555eb61e755c08a",
5    "extenUser": "KGpK4iWq",
6    "did": "0776707528",
7    "status": "Talking",
8    "direction": "Incoming",
9    "recordFile": "20220916103546022555eb61e755c08a-nikitaapi-972722776772-0776707528.mp3",
10   "isMuted": false,
11   "callStarted": 1663324548,
12   "isAnswered": true,
13   "callAnswered": 1663324555,
14   "currentCall": {
15     "callStarted": 1663324548,
16     "calldurationinterval": 1663324548,
17     "callAnswered": 1663324555,
18     "answered": 1,
19     "callername": "Queue Testing",
20     "callerphone": "972722776772",
21     "outgoingcallername": "",
22     "outgoingcallerphone": "",
23     "callstatus": "Talking",
24     "customdata": {
25     }
26     "direction": "Incoming",
27     "ivrid": "20220916103546022555eb61e755c08a",
28     "recording": {
29       "Filename": "20220916103546022555eb61e755c08a-nikitaapi-972722776772-0776707528.mp3",
30       "Options": "",
31       "ApproximateURL": "",
32       "IsMuted": 0
33     }
34     "did": "0776707528",
35     "relatedIvrUniqueIDs": [
36     ]
37     "callType": "Incoming",
38     "originalCallerID": "0722776772",
39     "originalCallerName": "Nikita Karpushin new",
40     "actualDialedNumber": "0776707528",
41     "channel": "SIP/KGpK4iWq-00000007",
42     "channel2": "SIP/ProviderProxy05-00000006",

```

```

43     "isSpied": false,
44     "isInternal": false,
45     "ip": "185.138.169.59",
46     "extenUser": "KGpK4iWq",
47     "number": 89312
48 }
49 }

```

## Hang up Phase

```

1  {
2    "phone": "972722776772",
3    "callerName": "Queue Testing",
4    "ivrid": "20220916103546022555eb61e755c08a",
5    "extenUser": "KGpK4iWq",
6    "did": "0776707528",
7    "status": "Talking",
8    "direction": "Incoming",
9    "recordFile": "20220916103546022555eb61e755c08a-nikitaapi-972722776772-0776707528.mp3",
10   "isMuted": false,
11   "callStarted": 1663324548,
12   "isAnswered": true,
13   "callAnswered": 1663324555,
14   "currentCall": {
15     "callStarted": 1663324548,
16     "calldurationinterval": 1663324548,
17     "callAnswered": 1663324555,
18     "answered": 1,
19     "callername": "Queue Testing",
20     "callerphone": "972722776772",
21     "outgoingcallername": "",
22     "outgoingcallerphone": "",
23     "callstatus": "Talking",
24     "customdata": {
25     }
26     "direction": "Incoming",
27     "ivrid": "20220916103546022555eb61e755c08a",
28     "recording": {
29       "Filename": "20220916103546022555eb61e755c08a-nikitaapi-972722776772-0776707528.mp3",
30       "Options": "",
31       "ApproximateURL": "",
32       "IsMuted": 0
33     }
34     "did": "0776707528",
35     "relatedIvrUniqueIDs": [
36     ]
37     "callType": "Incoming",
38     "originalCallerID": "0722776772",
39     "originalCallerName": "Nikita Karpushin new",
40     "actualDialedNumber": "0776707528",
41     "channel": "SIP/KGpK4iWq-000000007",
42     "channel2": "SIP/ProviderProxy05-000000006",
43     "isSpied": false,
44     "isInternal": false,
45     "ip": "185.138.169.59",
46     "extenUser": "KGpK4iWq",
47     "number": 89312,
48     "cause": "Normal hangup"
49   }
50 }

```

## Client Response (POST-JSON Only)

| Field Name | Description   | Type    | Remarks   |
|------------|---|---------|-----------|
| STATUS     | Only accepts "OK".<br>Any other value will indicate that there was an error on the client side.<br>Notice: "OK" in CAPITAL letters. | String  | Mandatory |
| URL        | A link to the caller's contact details page out of your business information software(CRM).   | String  | Optional  |
| CLIENTNAME | The caller's name out of your CRM.  | String  | Optional  |
| TOTAL      | How many results were found in your CRM.  | Integer | Optional  |
| COMPANY    | The caller's company name out of your CRM.  | String  | Optional  |

## JSON Response Example

```
1  {
2    "STATUS": "OK",
3    "URL": "https://www.yourdomain.com/contact_search.asp?user_phone=0722776772",
4    "CLIENTNAME": "John Doe",
5    "TOTAL": 1,
6    "COMPANY": "Voicenter"
7  }
```

# CDR Notification System API

The CDR Notification System API lets you store all of your business telephony information, such as call details records (CDRs), and manage it at any time directly from your database or your organizational CRM system. Our Cloud will send you an HTTP requests that contain detailed information for each call you made or received.



## GENERAL DESCRIPTION

If you wish to store and document the call detail records including the url for the call recording in your client's page inside your CRM system this is the API for you.

This api will also enable you to do further automated processes such as:

1. If you are using a queue and once you receive a call detail record when a client abandoned the call while waiting in the queue you may send an sms or a whatsapp to the client and/or popup a notification to the call center manager to callback the client.
2. Based on the received information you may analyze your call center performance as part of organizational business logic:

You will be able to exam which agents answer the most incoming calls.

Which agents produce the most answered outgoing calls.

In case you are performing outgoing calls to a variety of countries you can analyze which countries have the best answer ratio.

You will be able to view what are you peak hours in which you receive the biggest amount of incoming calls, and much more.

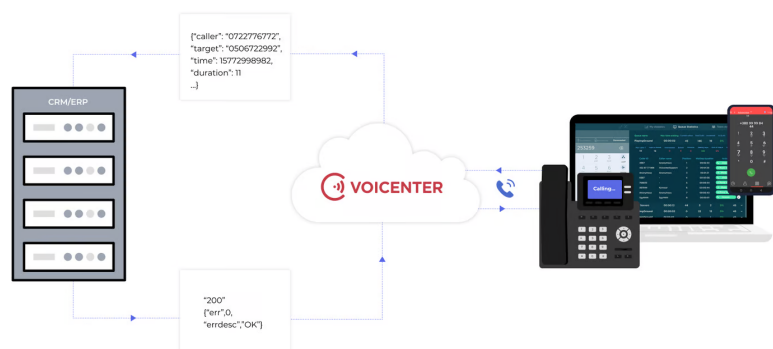
3. Along with our powerful AI processor you will be analyze your calls in more advanced manner, such as:

Receive the call's transcript to allow you to track key moments in the call easily.

You could view the emotions expressed in the call by your agents and your clients and their positive/negative emotional volatility.

The entire info from our AI processor will allow you to measure your support agent's or salesmen much more efficiently coupled with the info from your CRM (amount of ticket closed/leads converted).

## How Does The API Work?



1. A call is made to/from your Voicenter call center. After the call has ended, Voicenter converts the call data into a CDR.
2. Voicenter will send an HTTP request with your call data (CDR) to a Web-Service that was provided to us.
3. HTTP response will be sent back to Voicenter (HTTP response status code: 200 OK), in order to acknowledge us that the data was received.

## Request Parameters

- 1. POST-JSON
- 2. XML-RPC
- 3. Application/x-www-form-urlencoded

Response Parameters

- 1. POST-JSON
- 2. XML-RPC
- 3. Application/x-www-form-urlencoded

Voicenter Request - CDR Parameters

| Field       | Description   | Example  |
|-------------|---|--|
| caller      | Displays the caller’s phone number.<br>The phone number that will show to the call destination.   | "caller":"0722776772"  |
| target      | Displays the destination of the call.<br>Can be a phone number or the extension SIP code.<br>The phone number value will be sent with the international country prefix.   | "target":"SIPSIP" / "target":"972722776772"                                    |
| time        | Displays the time that that the call was made in Epoch time (Israel time zone).   | "time":1536855354  |
| duration    | Displays the duration of the call(seconds).<br>For regular incoming calls this duration does not include the ringing duration, only the actual time of the conversation that was made.<br>For calls of type queue this parameter will include also the waiting time of the call in the queue. | "duration":33  |
| ivruniqueid | Displays the ID code of the specific call.  | "ivruniqueid":"20200720095iMliIHJs"  |
| type        | Displays the type of Call.<br>For example: if it is an incoming/outgoing call?<br>There are several call types(More details are in the next pages).   | "type":"Incoming Call" / "type":"Extension Outgoing" / "type":"Click2Callleg1" |
| status      | Displays what happened with the specific call?<br>If the call had 2 "legs", this field will show the status of the leg 2 part of the call call(call types: Click2call, Automatic Dialer calls).<br>There are several call statuses(More details are below.                                    | "status":"ANSWER" / "status":"ABANDONE" / "status":"TE"                        |



| Field               | Description  | Example   |
|---------------------|--|---|
| leg1DialStatusName  | If the call had 2 legs, this field will show the status of the leg1 part of the call(call types: Click2call, Automatic Dialer calls). There are several call statuses(More details are in the next pages).   | "leg1DialStatusName": "ANSWER"/<br>"leg1DialStatusName": "ABANDONE" /<br>"leg1DialStatusName": "TE"                           |
| targetextension     | Displays the extension SIP code that answered the incoming call.<br>Sometimes this value will be identical to the "target" field.<br>There are cases that the incoming call is received not directly to the extension, so the "target" can display different values. | "targetextension": "SIPSIP"   |
| callerextension     | Displays the extension SIP code that the call was dialed from.<br>This value is different from the "caller" field.<br>In the "caller" field we display the actual number that will appear to the call destination.   | "callerextension": "SIPSIP"   |
| did                 | Displays the origin phone number that the caller called to.<br>This parameter will show value only on incoming calls.  | "did": "0722776772"   |
| queueid             | If the call was directed to a queue service, it displays the queue code ID.<br>The value of the field will be null in case there is on queue.  | "queueid": 1234567  |
| queuename           | If the call was directed to a queue service, it displays the queue name.   | "queuename": "Service Queue"  |
| record              | Displays the URL link to the call recording.   | "record": "https://cpanel.voicenter.co.il/CallsHistory/PlayRecord/2022071916331502463359186d8ab5f5-aws-APIAPI-0722776772.mp3" |
| price               | Displays the total price of the call in ILS cents (Agorot).  | "price": 7  |
| dialtime            | Displays the ringing duration of the call(seconds).<br>Not include the actual conversation duration.   | "dialtime": 23  |
| representative_name | Displays the Voicenter user name that the specific call was associated with.<br>If a user was not logged in when the call was made, then this field will show the user name that the extension\ DID is associated with.  | "representative_name": "Walter Melon"   |
|                     |  |   |

| Field                    | Description  | Example   |
|--------------------------|--|---|
| representative_code      | Displays the Voicenter user ID code that the specific call was associated with.<br>If a user was not logged in when the call was made, then this field will show the user name that the extension\ DID is associated with.   | "representative_code": "9996"                       |
| targetextension_name     | Displays the Voicenter extension name that answered the specific call.   | "targetextension_name": "Walter Melon"              |
| callerextension_name     | Displays the Voicenter extension name that this specific call was made from.   | "callerextension_name": "Walter Melon"              |
| target_country           | Displays the country name that this outgoing call was made to.   | "target_country": "Israel"                          |
| caller_country           | Displays the country name that this incoming call was made from.   | "caller_country": "Israel"                          |
| seconds_waiting_in_queue | This field will only be sent, if the specific call was directed to a queue.<br>It displays the duration (seconds) that the caller waited in the queue.   | "seconds_waiting_in_queue": 5                       |
| OriginalIvrUniqueID      | This field will only be sent if a specific call was related to another call.<br>It displays the origin call code ID – "ivruniqueid".<br>For example, when calls are transferred in the organization between representatives. | "OriginalIvrUniqueID":<br>"201809131730110122A1212" |
| DepartmentName           | Displays the department\ account name(as it appears in Voicenter) that the call was associated with.   | "DepartmentName": "Voicenter Account"               |
| DepartmentID             | Displays the department\ account ID(as it appears in Voicenter) that the call was associated with.   | "DepartmentId": 123456                              |
| TopDepartmentName        | If the given account has a hierarchy, this field will display the name of the top account in the hierarchy.  | "TopDepartmentID": "Voicenter Top Account"          |
| TopDepartmentID          | If the given account has a hierarchy, this field will display the ID of the top account in the hierarchy.  | "TopDepartmentID": 7654321                          |
|                          |  |   |

| Field          | Description  | Example   |
|----------------|--|---|
| IVR            | <p>An array type field. Displays the stages in the IVR that an incoming call went through.</p> <p>Fields:</p> <p>"layer_id" - The IVR layer Voicenter unique ID.</p> <p>"layer_name" – The IVR layer name as it appears in the Voicenter control panel (CPanel).</p> <p>"DTMF" – If the caller pressed on a digit\s. If the caller did not press an any digit the default value will be "0".</p> <p>"LayerNumber" - The IVR layer ID.</p> <p>"dtmf_order" – The layer order that the call went through in the IVR.</p> | <pre>"IVR": "[{"layer_id":1234567, "layer_name":"LayerName", "layer_number":6, "Dtmf":0, "dtmf_order":1 }]"</pre>   |
| campaignName   | <p>In the case of an auto-dialer call, this field will display the name of the specific auto-dialer campaign through which the call was made. This field will only be sent on auto-dialer calls.</p>   | <pre>"campaignName": "Leads 2024",</pre>  |
| campaignCode   | <p>In the case of an auto-dialer call, this field will display the API code of the specific auto-dialer campaign through which the call was made. This information is mainly relevant for integration usage. This field will only be sent in auto-dialer calls.</p>  | <pre>"campaignCode": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXX",</pre>   |
| dialerData     | <p>In the case of an auto-dialer call, an object containing information detailing the campaign's details through which the call was made. This object will only be sent in auto-dialer calls.</p>  | <pre>"dialerData": { "CampaignID": 123456789, "CampaignTypeID": 5, "CampaignTypeName": "Automatic", "CallID": "1234567", "StatusName": "ANSWER", "Phone": "972722776772", "CallInserted": "2024-11-25T16:10:21.790Z", "RetryNum": 2, "RetryAmount": 3 }</pre> |
| campaignID     | <p>In the case of an auto-dialer call, this field will display the ID of the specific auto-dialer campaign through which the call was made. This field will only be sent in auto-dialer calls.</p>   | <pre>"campaignID": 123456789</pre>  |
| CampaignTypeID | <p>In the case of an auto-dialer call, this field will display the specific auto-dialer campaign type identifier through which the call was made. The field contains one of two options: 1. 5 (the service first dials to the representative and then the customer), 2. 3 (the service first dials to the customer and then returns to the representative). This field will only be sent in auto-dialer calls.</p>   | <pre>"CampaignTypeID": 5,</pre>   |

| Field            | Description  | Example                                     |
|------------------|--|---|
| campaignTypeName | In the case of an auto-dialer call, this field will display the specific auto-dialer campaign type through which the call was made. The field contains one of two options: 1. Automatic (the service first dials to the representative and then the customer), 2. IVR (the service first dials to the customer and then returns to the representative). This field will only be sent in auto-dialer calls.   | "campaignTypeName": "Automatic",            |
| CallID           | In the case of an auto-dialer call, this field will display the target ID in the campaign. Note, that this field value is not equal to the ivruniqueid field. Each target has an ID in the campaign and there can be multiple call attempts to the same target (multiple ivruniqueids). This field will only be sent in auto-dialer calls.   | "CallID": "1083439851",                     |
| StatusName       | In the case of an auto-dialer call, this field will display the call answer status. For more details, see the table below. This field will only be sent in auto-dialer calls.  | "StatusName": "ANSWER",                     |
| Phone            | In the case of an auto-dialer call, this field will display the phone number of the call destination as was sent to the service. This field will only be sent in auto-dialer calls.  | "Phone": "972722776772",                    |
| CallInserted     | In the case of an auto-dialer call, this field will display the time the destination was entered into the service. This field will only be sent in auto-dialer calls.  | "CallInserted": "2024-11-25T16:10:21.790Z", |
| RetryNum         | In the case of an auto-dialer call, this field will display the current number of attempts to dial the destination. Within the service, you can define the number of attempts for the service to dial the destination. The service will stop dialing the destination either when the destination answers the call or when the service has dialed the maximum number of attempts that have been defined. This field will only be sent in auto-dialer calls. | "RetryNum": 2,                              |
|                  |  |   |

| Field       | Description  | Example  |
|-------------|--|--|
| RetryAmount | In the case of an auto-dialer call, this field will display the maximum number of attempts set to dial the destination. Within the service, you can set the number of attempts the service will make to dial the destination. The service will stop dialing the destination either when the destination answers the call or when the service has dialed the maximum number of attempts set. This field will only be sent in auto-dialer calls. | "RetryAmount": 3   |
| aiData      | JSON object that shows the entire AI data for the call. Contains these objects: insights, emotions and transcript. The service is currently supported in Hebrew and English. To add additional languages, please send a request to the Voicenter support department.   |  |
| insights    | This object contains the data analysis of the call. It contains questions(AI prompts), the call summary, and the participants' analysis.   | <pre>"insights":{ "questions":[ { "key":"category", "question":"Categorize the call to exactly one of the possible categories:"technical issue","billing issues","general support issues", "answer":"technical issue" } ], "participants":{ "caller":{"name":null, "personality_traits":[] }, "callee":{"name":"Steve", "personality_traits":[] } }, "summary":"The caller had clash phone service issue. Steve tried troubleshooting steps like restarting the phone and resetting network settings, which solved the issue." }</pre> |
| questions   | This array contains an object of the questions(prompts) that the AI engine will analyze. It contains: key - the name of the current category. question - the prompt question. data_type - the classification of the returned value type of the AI analysis(text, number, array, etc.). answer - the return value of the AI service analysis. The questions can be managed in the Cpanel in Account Management advanced settings.               | <pre>{ "questions": [ { "key": "category", "question": "Categorize the call to exactly one of the possible categories: "technical issue", "sales", "other issues"", "data_type": "string", "answer": "technical issue" }, { "key": "issue", "question": "What was the technical issue?", "data_type": "string", "answer": "The caller was experiencing issues with their phone service." } ] }</pre>   |
| key         | The category name of each question.  | "key": "call_type"   |
| question    | The prompt question for the AI engine to analyze.  | "question": "Categorize the call to exactly one of the possible categories: "technical issue", "billing issues", "general support issues", "sales", "other issues""  |
| data_type   | The data type that the response answer field will be. The data types are: 1-boolean, 2-string, 3-number, 4-json array, 5-json object list, 6-json object   | "data_type": "string"  |

| Field              | Description  | Example   |
|--------------------|--|---|
| answer             | The question returned value.   | "answer": "technical issue"   |
| participants       | This object analyzes the caller and the callee's personality traits. It contains objects for the caller and the callee with each name(if recognized) and each person's traits.   | { "participants": { "caller": { "name": null, "personality_traits": [] }, "callee": { "name": "Steve", "personality_traits": [ "professional", "patient", "trying to explain clearly" ] } } }   |
| name               | The name of the call\callee.   | "name": "Steve"   |
| personality_traits | An array that contains the personal characteristics of the caller and the callee.  | "personality_traits": [ "professional", "patient", "trying to explain clearly" ]  |
| summary            | Summarize the transcript of the conversation in a few lines.   | "summary": "The customer wanted to exchange virtual numbers and purchase a new virtual number. The representative helped the customer change 7 virtual numbers and purchase 1 new virtual number, a total of 8 new numbers. The representative explained to the customer the costs involved and the customer approved the costs. In the end, the representative confirmed that he would email the new numbers to the customer." |
| emotions           | This object contains the emotional analysis of the caller and callee. The AI engine returns each sentence and the emotion associated with it. The size of this object can vary. It depends on the length of the call and the ability of the AI engine to recognize each sentence's emotions(there may be sentences in which no distinct emotion was detected). | { "emotions":{ "sentences":[ { "sentence_id":28, "emotion":"frustrated", "emotion_direction":-1, "confidence_emotion":0.88, "intensity_emotion":0.75, "personality_trait":"impatient", "confidence_trait":0.85 }, { "sentence_id":34, "emotion":"professional", "emotion_direction":1, "confidence_emotion":0.92, "intensity_emotion":0.85, "personality_trait":"helpful", "confidence_trait":0.88 } ] } }                      |
| sentences          | This array contains objects of each sentence that the AI engine succeeded in recognizing an emotion.   | { "sentence_id": 28, "emotion": "frustrated", "emotion_direction": -1, "confidence_emotion": 0.88, "intensity_emotion": 0.75, "personality_trait": "impatient", "confidence_trait": 0.85 }, { "sentence_id": 34, "emotion": "professional", "emotion_direction": 1, "confidence_emotion": 0.92, "intensity_emotion": 0.85, "personality_trait": "helpful", "confidence_trait": 0.88 },  |
| sentence_id        | Each transcribed line of the call has an ID starting from 0.   | "sentence_id": 56   |
| emotion            | The recognize emotion of the sentence.   | "emotion": "frustrated"   |
|                    |  |   |

| Field              | Description   | Example  |
|--------------------|---|--|
| emotion_direction  | This field gives a numeric direction of sentence emotion: 1: positive, -1: negative, 0: neutral.  | "emotion_direction": -1  |
| confidence_emotion | The AI engine numerically evaluates (between 0-1) how likely the emotion it detected is correct.  | "confidence_emotion": 0.92   |
| intensity_emotion  | The AI engine numerically evaluates (between 0-1) the level of intensity of the detected emotion. | "intensity_emotion": 0.8   |
| personality_trait  | The recognized personality trait of the sentence.   | "personality_trait": "patient"   |
| confidence_trait   | The AI engine numerically evaluates (between 0-1) how likely the trait it detected is correct.    | "confidence_trait": 0.82   |
| transcript         | The full call transcript. It contains all sentences separated into objects.                       | { "speaker": "Speaker0", "text": "H: Good afternoon", "startTime": 1.87, "endTime": 1.89, "sentence_id": 0 }, { "speaker": "Speaker0", "text": "thank you for calling voice and support", "startTime": 3, "endTime": 5.2, "sentence_id": 1 } |
| speaker            | Speaker0 is the representative. Speaker1 is the client.   | { "speaker": "Speaker0", "text": "how can i assist you today", "startTime": 8, "endTime": 9.64, "sentence_id": 3 }, { "speaker": "Speaker1", "text": "hi i'm having some issue", "startTime": 13.46, "endTime": 13.48, "sentence_id": 4 }    |
| text               | The transcript text of the current sentence.  | "text": "how can i assist you today"   |
| startTime          | The start time of the current sentence. In seconds, since the conversation started.               | "startTime": 8   |
| endTime            | The end time of the current sentence. In seconds, since the conversation started.                 | "endTime": 9.64  |
| sentence_id        | Each transcribed line of the call has an ID starting from 0.                                      | "sentence_id": 3   |

## Call Types

| CDR Type Name | Description  |
|---------------|--|
| Incoming Call | A regular incoming call that was dialed to a phone number and ended up in the IVR or an extension (did not end in a queue, which will make it type 4). |
| CC            | A call that was made through a calling card (Access number) service.   |
|               |  |

| CDR Type Name          | Description   |
|------------------------|---|
| Extension Outgoing     | A regular outgoing call (manually dialed from the phone).   |
| Queue                  | An incoming call that was received by a queue.  |
| Click2Call leg1        | A call that was made by click2call (Not by Dialer), leg 1 of the call.<br>Leg 1 - the initial connection of the call to the extension.<br>This is the “Phone” parameter in the Click2call API.<br>This part will contain info about the initial connection of the call.   |
| Click2Call leg2        | A call that was made by click2call (Not by Dialer), leg 2 of the call.<br>Leg 2 - the actual call that is being made to the destination.<br>This is the “Target” parameter in the Click2call API.<br>This part will contain the call details and call recording.  |
| VoiceMail              | A call that was answered by Voicenter voicemail.  |
| Callference            | A call that was made through Voicenter callference service.   |
| XferCDR                | A call that manually transferred from an extension to a DID/another extension.  |
| ProductiveCall Leg1    | A "leg 1" Agents Auto Dialer calls.<br>Leg1 - the initial connection of the call to the extension.<br>This part will contain info about the initial connection of the call.   |
| ProductiveCall Leg2    | A "leg 2" Agents Auto Dialer calls.<br>Leg 2 -the actual call that is being made to the destination.<br>This part will contain the call details and call recording.   |
| Scrubber               | A call that was made through Voicenter's Scrubber service.  |
| Click 2 IVR            | "Leg1" Predictive Dialer calls.<br>Leg1 - the initial connection of the call to the destination.<br>This part will contain info about the initial connection of the call.   |
| Click 2 IVR Incoming   | This is the "Leg 2" Predictive Dialer call.<br>After the initial Leg1 (Click 2 IVR) call was answered, this new leg is dialing the IVR (a layer in Voicenter telephony menu) and Leg 1 is connected to the representatives.<br>This call type is for all Click 2 IVR that did not enter a queue.<br>If the call enters a queue service, the type will be type 15. This part will contain the call details and call recording. |
| Click 2 Queue Incoming | This is the "Leg 2" Predictive Dialer call.<br>After the initial Leg1 (Click 2 IVR) call was answered, this new leg is dialing the IVR (a layer in Voicenter telephony menu) and Leg 1 is connected to the representatives.<br>If the call enters a queue, then the call status will be - Click 2 Queue. Else, the call type will be type 14. This part will contain the call details and call recording.                     |
| FaxCdr                 | A call that was made through Voicenter's internal outgoing fax service. *Not released yet.  |
| Attended CDR leg1      | A call that was transferred with consultation.  |
|                        |   |



| CDR Type Name     | Description   |
|-------------------|---|
| Attended CDR leg2 | A call that was transferred with consultation.<br>This type will only be made in a case of an incoming call that was answered by representative "A", "A" will then put the call on hold and make another call (consult) to another person - "B".<br>Afterward, "A" will transfer the initial call to "B". The "Leg 2" is the part of the call between "B" and the initial caller. |
| Auto forward      | A call that was automatically transferred from an extension to a DID (usually representatives configure their phones manually to transfer calls when they are not available).   |

## Leg 1 & Leg 2 Call Types

In some call types, each call consists of two parts arranged in chronological order:

- Click2Call** – The first part of the call (Leg1) is the initial connection to the user extension. If the initial connection was established, then the actual call to the destination is being made (Leg2).
- Productive Dialer** – The first part of the call (Leg1) is the initial connection to the user extension. If the initial connection was established, then the actual call to the destination is being made (Leg2).
- Predictive Dialer** – The first part of the call (Leg1) trying to call the destination. If the initial connection was established to the call destination, then the call is connected to your call center (Leg2).

The Leg2 call parts are often used because it is the actual call to the destination. Sometimes, in the case of Leg2 call part that failed to connect to the destination (call status – NOTCALLED), the Leg1 part can show the reason that the call did not connect to the extension.

## Call Statuses

| CDR Status Name | Description   |
|-----------------|---|
| NOTDIALED       | Hang-up occurred before the call was made.  |
| ANSWER          | A call was answered.<br>A successful dial.<br>The caller reached the destination.<br>Whenever we receive an answer response signal, also when the call reached local voicemail service and etc. |
| BUSY            | Busy signal. The dial command reached its number but the number is busy.  |
| NOANSWER        | No answer. The dial command reached its number, the number rang for too long, then the dial timed out   |
| CANCEL          | A call is canceled. The dial command reached its number but the caller hung up before the call destination could answer.  |

| CDR Status Name | Description   |
|-----------------|---|
| ABANDONE        | When using Voicenter's queue service, this status will appear in several cases:<br>A caller hung up before the call destination could answer.<br>A caller while waiting in the queue, chose to exit from the queue.<br>A call has reached timeout in the queue. |
| VOEND           | Hang-up during IVR without actual dialing.<br>In this case, the caller waited in the IVR but hung up before the call rang in any extension.   |
| TE              | When an incoming call is directed to an IVR recording and afterward it configured to hung up the call.  |
| NOTCALLED       | A Leg2 Click2Call was not called.<br>When using a click2call service and the Leg1 initial stage was not successful (not answered).  |
| VOICEMAIL       | Call entered to Voicenter voicemail service.  |

Call Error Types

There maybe be time where while trying to make an outgoing call to certain destination you will receive an error. These error responses display cases when there was a problem connecting to the target destination. It is mainly used for Voicenter internal Tracking. The errors you may receive are as followed:

| Error Types | Error Description   |
|-------------|---|
| CONGESTION  | Congestion. This status is usually a sign that the dialed number is not recognized. |
| CHANUNAVAIL | Channel unavailable. On SIP, peer may not be registered.                            |
| INVALIDARGS | Error parsing dial command arguments.   |
| SSWPREAUTH  | SSW outgoing call cancel before actual dial.  |

Client Response

It is important that we get a feedback to understand that all data was received. In case, we do not receive a valid response we will re-send the CDR data that we failed to send. The amount of retries and the time intervals between retries are set on Voicenter side.

| Name    | Type    | Exmple | Description  |
|---------|---------|--------|--|
| Err     | Integer | 0      | Error codes:<br>0 – OK<br>1 – Parse error<br>2 – Application error |
| Errdesc | String  | OK     | Error description  |

JSON Example (Outgoing Call):

```
|
```

```

1  {
2      "caller": "0722776772",
3      "target": "0501234567",
4      "time": 1595960350,
5      "duration": 11,
6      "ivruniqueid": "2020072818dcDHFJcc804",
7      "type": "Extension Outgoing",
8      "status": "ANSWER",
9      "targetextension": "",
10     "callerextension": "SIPSIP",
11     "did": "",
12     "queueid": 0,
13     "queuename": "",
14     "record": "https://cpanel.voicenter.co.il/CallsHistory/PlayRecord/2020072818dcDHFJcc804-aws.mp3",
15     "price": 0,
16     "dialtime": 2,
17     "representative_name": "User 1",
18     "representative_code": "12345678",
19     "targetextension_name": "",
20     "callerextension_name": "",
21     "target_country": "",
22     "caller_country": "",
23     "DepartmentID": 12345678,
24     "DepartmentName": "Voicenter Sales Department",
25     "TopDepartmentID": 87654321,
26     "TopDepartmentName": "Voicenter Account"
27 }

```

In this example, a user (User 1) made an outgoing call from extension (SIPSIP), the call was dialed manually from the extension (not through Auto-Dialer or Click2call services).

The user called the phone number - "0501234567", the phone number that appeared to the call destination was - "0722776772".

This call was answered ("ANSWER").

The call rang 2 seconds, and the duration of the conversation was 11 seconds.

## JSON Request Example (Incoming Call):

```

1  {
2      "caller": "0501234567",
3      "target": "",
4      "time": 1595333610,
5      "duration": 20,
6      "ivruniqueid": "202007211213270124c",
7      "type": "Queue",
8      "status": "ANSWER",
9      "targetextension": "",
10     "callerextension": "",
11     "did": "0722776772",
12     "queueid": 123456789,
13     "queuename": "Sales Queue",
14     "record": "https://cpanel.voicenter.co.il/CallsHistory/PlayRecord/202007211213270124c-aws.mp3",
15     "price": 0,
16     "dialtime": 0,
17     "representative_name": "Admin User",
18     "representative_code": "12345679",
19     "targetextension_name": "",
20     "callerextension_name": "",
21     "target_country": "",
22     "caller_country": "Israel",
23     "seconds_waiting_in_queue": 20,
24     "IVR": [
25         {
26             "layer_id": 1234,

```

```

27     "layer_name": "Main IVR",
28     "layer_number": 0,
29     "Dtmf": 2,
30     "dtmf_order": 1
31 }
32 {
33     "layer_id": 4321,
34     "layer_name": "Sales Department",
35     "layer_number": 2,
36     "Dtmf": 0,
37     "dtmf_order": 2
38 }
39 ]
40 }

```

In this example, an incoming call was made from - "0501234567", the caller called the phone number "0722776772". This call went through the IVR, first, on the "Main IVR" layer which then the caller pressed "2" and was directed to the second layer – "Sales Department".

This call rang in the queue service ("Queue") name - "Sales Queue".

The caller waited in the queue for 20 seconds and decided to hang-up the call ("ABANDONE").

## Request Example With AI

```

1  {
2    "IVR": [
3      {
4        "layer_id": 196285,
5        "layer_name": "0747809282",
6        "layer_number": 4,
7        "Dtmf": 0,
8        "dtmf_order": 1
9      }
10   ]
11   "caller": "0501234567",
12   "target": "SIP12SIP",
13   "time": 1721942255,
14   "duration": 333,
15   "ivruniqueid": "202407252117332121279029ab1d84",
16   "type": "Incoming Call",
17   "status": "ANSWER",
18   "targetextension": "SIP12SIP",
19   "callerextension": "",
20   "did": "0722776772",
21   "queueid": 0,
22   "queuename": "",
23   "record": "https://cpanel.voicenter.co.il/CallsHistory/PlayRecord/202407252117332121279029ab1d84-vc213.mp3.706",
24   "price": 0,
25   "dialtime": 11,
26   "representative_name": "Test 1",
27   "representative_code": "1882",
28   "targetextension_name": "Test 1",
29   "callerextension_name": "",
30   "target_country": "",
31   "caller_country": "Israel",
32   "aiData": {
33     "insights": {
34       "questions": [
35         {
36           "key": "category",
37           "question": "Categorize the call to exactly one of the possible categories: 'technical issue', 'billing",
38           "data_type": "string",
39           "answer": "technical issue"

```

```

40 }
41 {
42   "key": "issue",
43   "question": "What was the technical issue?",
44   "data_type": "string",
45   "answer": "The caller was experiencing issues with their Clash phone service, including dropped calls and low call quality."
46 }
47 {
48   "key": "issue_resolution",
49   "question": "Please mention in at least 3 sentences the steps taken to try and resolve the issue and how the agent helped.",
50   "data_type": "string",
51   "answer": "The agent tried the following steps to resolve the issue: 1) Asked the caller to restart the phone. 2) Offered to transfer the caller to a specialist if needed. 3) Provided a troubleshooting guide for common issues."
52 }
53 {
54   "key": "issue_resolution_confirmation",
55   "question": "Did the caller confirm the issue was solved?",
56   "data_type": "string",
57   "answer": "The caller confirmed that the call quality seemed a bit better after resetting the network settings."
58 }
59 {
60   "key": "issue_resolved",
61   "question": "Was the issue resolved? (true/false)",
62   "data_type": "boolean",
63   "answer": "false"
64 }
65 {
66   "key": "key_phrases_speaker0",
67   "question": "Provide an array with a list of up to 5 key phrases said by the first participant of the call.",
68   "data_type": "json array",
69   "answer": [
70     "thank you for calling voice and support",
71     "how can i assist you today",
72     "I'll do my best to help you resolve them",
73     "let me pull up your account",
74     "have you tried restarting your phone"
75   ]
76 }
77 {
78   "key": "key_phrases_speaker1",
79   "question": "Provide an array with a list of up to 5 key phrases said by the second participant of the call.",
80   "data_type": "json array",
81   "answer": [
82     "i'm having some issues with clash phone service",
83     "my calls keep dropping and the quality is really bad",
84     "it started a few days ago",
85     "it is mostly happening at home",
86     "I'm using an iPhone"
87   ]
88 }
89 {
90   "key": "keywords_speaker0",
91   "question": "Provide an array with a list of up to 20 keywords used by the first participant during the call.",
92   "data_type": "json object list",
93   "answer": [
94     {
95       "word": "thank",
96       "count": 3
97     },
98     {
99       "word": "account",
100      "count": 2
101     },
102     {
103       "word": "issue",
104       "count": 2
105     },
106     {

```

```

107         "word": "help",
108         "count": 2
109     }
110     {
111         "word": "reset",
112         "count": 2
113     }
114 ]
115 }
116 {
117     "key": "keywords_speaker1",
118     "question": "provide an array with a list of up to 20 keywords used by the second participant during the call.",
119     "data_type": "json object list",
120     "answer": [
121         {
122             "word": "issue",
123             "count": 2
124         }
125         {
126             "word": "call",
127             "count": 3
128         }
129         {
130             "word": "drop",
131             "count": 2
132         }
133         {
134             "word": "quality",
135             "count": 2
136         }
137         {
138             "word": "home",
139             "count": 1
140         }
141     ]
142 }
143 {
144     "key": "technical issues",
145     "question": "Identify recurring technical issues mentioned by customers. Provide an array.",
146     "data_type": "json array",
147     "answer": [
148         "Dropped calls",
149         "Poor call quality"
150     ]
151 }
152 {
153     "key": "topics",
154     "question": "Summarize the main 3 topics of the call. Provide an array.",
155     "data_type": "json array",
156     "answer": [
157         "Issues with Clash phone service",
158         "Troubleshooting steps",
159         "Creating a ticket for further investigation"
160     ]
161 }
162 {
163     "key": "troubleshooting",
164     "question": "Highlight successful troubleshooting steps and their frequency.",
165     "data_type": "string",
166     "answer": "The successful troubleshooting steps mentioned were: 1) Restarting the phone. 2) Checking network connection."
167 }
168 ]
169 "participants": {
170     "caller": {
171         "name": null,
172         "personality_traits": [
173

```

```

174     }
175     "callee":{
176         "name":"Steve",
177         "personality_traits":[
178             ]
179     }
180 }
181 "summary":"The caller, whose name is not mentioned, contacted Steve from voice and support regarding issue
182 }
183 "emotions":{
184     "questions":[
185     ]
186     "sentences":[
187         {
188             "sentence_id":21,
189             "emotion":"frustrated",
190             "emotion_direction":-1,
191             "confidence_emotion":0.92,
192             "intensity_emotion":0.85,
193             "personality_trait":"impatient",
194             "confidence_trait":0.9
195         }
196         {
197             "sentence_id":23,
198             "emotion":"frustrated",
199             "emotion_direction":-1,
200             "confidence_emotion":0.9,
201             "intensity_emotion":0.8,
202             "personality_trait":"impatient",
203             "confidence_trait":0.88
204         }
205         {
206             "sentence_id":28,
207             "emotion":"frustrated",
208             "emotion_direction":-1,
209             "confidence_emotion":0.88,
210             "intensity_emotion":0.75,
211             "personality_trait":"impatient",
212             "confidence_trait":0.85
213         }
214         {
215             "sentence_id":34,
216             "emotion":"professional",
217             "emotion_direction":1,
218             "confidence_emotion":0.92,
219             "intensity_emotion":0.85,
220             "personality_trait":"helpful",
221             "confidence_trait":0.88
222         }
223         {
224             "sentence_id":35,
225             "emotion":"neutral",
226             "emotion_direction":0,
227             "confidence_emotion":0.5,
228             "intensity_emotion":0.3,
229             "personality_trait":null,
230             "confidence_trait":0
231         }
232         {
233             "sentence_id":37,
234             "emotion":"helpful",
235             "emotion_direction":1,
236             "confidence_emotion":0.9,
237             "intensity_emotion":0.75,
238             "personality_trait":"patient",
239             "confidence_trait":0.85
240     }

```

```
241 {
242     "sentence_id":38,
243     "emotion":"cooperative",
244     "emotion_direction":1,
245     "confidence_emotion":0.88,
246     "intensity_emotion":0.7,
247     "personality_trait":"compliant",
248     "confidence_trait":0.82
249 }
250 {
251     "sentence_id":39,
252     "emotion":"neutral",
253     "emotion_direction":0,
254     "confidence_emotion":0.6,
255     "intensity_emotion":0.4,
256     "personality_trait":null,
257     "confidence_trait":0
258 }
259 {
260     "sentence_id":40,
261     "emotion":"frustrated",
262     "emotion_direction":-1,
263     "confidence_emotion":0.92,
264     "intensity_emotion":0.8,
265     "personality_trait":"impatient",
266     "confidence_trait":0.88
267 }
268 {
269     "sentence_id":41,
270     "emotion":"helpful",
271     "emotion_direction":1,
272     "confidence_emotion":0.93,
273     "intensity_emotion":0.82,
274     "personality_trait":"professional",
275     "confidence_trait":0.9
276 }
277 {
278     "sentence_id":42,
279     "emotion":"helpful",
280     "emotion_direction":1,
281     "confidence_emotion":0.9,
282     "intensity_emotion":0.75,
283     "personality_trait":"patient",
284     "confidence_trait":0.85
285 }
286 {
287     "sentence_id":43,
288     "emotion":"neutral",
289     "emotion_direction":0,
290     "confidence_emotion":0.5,
291     "intensity_emotion":0.3,
292     "personality_trait":null,
293     "confidence_trait":0
294 }
295 {
296     "sentence_id":45,
297     "emotion":"helpful",
298     "emotion_direction":1,
299     "confidence_emotion":0.92,
300     "intensity_emotion":0.8,
301     "personality_trait":"patient",
302     "confidence_trait":0.88
303 }
304 {
305     "sentence_id":46,
306     "emotion":"helpful",
307     "emotion_direction":1,
```



```
308         "confidence_emotion":0.93,
309         "intensity_emotion":0.82,
310         "personality_trait":"professional",
311         "confidence_trait":0.9
312     }
313     {
314         "sentence_id":47,
315         "emotion":"neutral",
316         "emotion_direction":0,
317         "confidence_emotion":0.6,
318         "intensity_emotion":0.4,
319         "personality_trait":null,
320         "confidence_trait":0
321     }
322     {
323         "sentence_id":48,
324         "emotion":"cooperative",
325         "emotion_direction":1,
326         "confidence_emotion":0.88,
327         "intensity_emotion":0.7,
328         "personality_trait":"compliant",
329         "confidence_trait":0.82
330     }
331     {
332         "sentence_id":49,
333         "emotion":"appreciative",
334         "emotion_direction":1,
335         "confidence_emotion":0.9,
336         "intensity_emotion":0.75,
337         "personality_trait":"polite",
338         "confidence_trait":0.85
339     }
340     {
341         "sentence_id":50,
342         "emotion":"neutral",
343         "emotion_direction":0,
344         "confidence_emotion":0.5,
345         "intensity_emotion":0.3,
346         "personality_trait":null,
347         "confidence_trait":0
348     }
349     {
350         "sentence_id":51,
351         "emotion":"helpful",
352         "emotion_direction":1,
353         "confidence_emotion":0.92,
354         "intensity_emotion":0.8,
355         "personality_trait":"patient",
356         "confidence_trait":0.88
357     }
358     {
359         "sentence_id":52,
360         "emotion":"neutral",
361         "emotion_direction":0,
362         "confidence_emotion":0.5,
363         "intensity_emotion":0.3,
364         "personality_trait":null,
365         "confidence_trait":0
366     }
367     {
368         "sentence_id":53,
369         "emotion":"neutral",
370         "emotion_direction":0,
371         "confidence_emotion":0.5,
372         "intensity_emotion":0.3,
373         "personality_trait":null,
374         "confidence_trait":0
```

```
375     }
376     {
377         "sentence_id":54,
378         "emotion":"neutral",
379         "emotion_direction":0,
380         "confidence_emotion":0.5,
381         "intensity_emotion":0.3,
382         "personality_trait":null,
383         "confidence_trait":0
384     }
385     {
386         "sentence_id":56,
387         "emotion":"helpful",
388         "emotion_direction":1,
389         "confidence_emotion":0.92,
390         "intensity_emotion":0.8,
391         "personality_trait":"patient",
392         "confidence_trait":0.88
393     }
394     {
395         "sentence_id":57,
396         "emotion":"patient",
397         "emotion_direction":1,
398         "confidence_emotion":0.9,
399         "intensity_emotion":0.75,
400         "personality_trait":"helpful",
401         "confidence_trait":0.85
402     }
403     {
404         "sentence_id":58,
405         "emotion":"neutral",
406         "emotion_direction":0,
407         "confidence_emotion":0.6,
408         "intensity_emotion":0.4,
409         "personality_trait":null,
410         "confidence_trait":0
411     }
412     {
413         "sentence_id":59,
414         "emotion":"cooperative",
415         "emotion_direction":1,
416         "confidence_emotion":0.88,
417         "intensity_emotion":0.7,
418         "personality_trait":"compliant",
419         "confidence_trait":0.82
420     }
421     {
422         "sentence_id":60,
423         "emotion":"neutral",
424         "emotion_direction":0,
425         "confidence_emotion":0.6,
426         "intensity_emotion":0.4,
427         "personality_trait":null,
428         "confidence_trait":0
429     }
430     {
431         "sentence_id":62,
432         "emotion":"helpful",
433         "emotion_direction":1,
434         "confidence_emotion":0.92,
435         "intensity_emotion":0.8,
436         "personality_trait":"patient",
437         "confidence_trait":0.88
438     }
439     {
440         "sentence_id":63,
441         "emotion":"neutral",
```

```
442     "emotion_direction":0,
443     "confidence_emotion":0.6,
444     "intensity_emotion":0.4,
445     "personality_trait":null,
446     "confidence_trait":0
447 }
448 {
449     "sentence_id":65,
450     "emotion":"hopeful",
451     "emotion_direction":0,
452     "confidence_emotion":0.88,
453     "intensity_emotion":0.7,
454     "personality_trait":"patient",
455     "confidence_trait":0.82
456 }
457 {
458     "sentence_id":66,
459     "emotion":"empathetic",
460     "emotion_direction":1,
461     "confidence_emotion":0.9,
462     "intensity_emotion":0.75,
463     "personality_trait":"understanding",
464     "confidence_trait":0.85
465 }
466 {
467     "sentence_id":67,
468     "emotion":"helpful",
469     "emotion_direction":1,
470     "confidence_emotion":0.92,
471     "intensity_emotion":0.8,
472     "personality_trait":"patient",
473     "confidence_trait":0.88
474 }
475 {
476     "sentence_id":68,
477     "emotion":"neutral",
478     "emotion_direction":0,
479     "confidence_emotion":0.5,
480     "intensity_emotion":0.3,
481     "personality_trait":null,
482     "confidence_trait":0
483 }
484 {
485     "sentence_id":69,
486     "emotion":"confident",
487     "emotion_direction":1,
488     "confidence_emotion":0.93,
489     "intensity_emotion":0.88,
490     "personality_trait":"professional",
491     "confidence_trait":0.93
492 }
493 {
494     "sentence_id":70,
495     "emotion":"helpful",
496     "emotion_direction":1,
497     "confidence_emotion":0.95,
498     "intensity_emotion":0.85,
499     "personality_trait":"customer service oriented",
500     "confidence_trait":0.92
501 }
502 {
503     "sentence_id":72,
504     "emotion":"urgent",
505     "emotion_direction":-1,
506     "confidence_emotion":0.9,
507     "intensity_emotion":0.8,
508     "personality_trait":"impatient",
```

```

509         "confidence_trait":0.88
510     }
511     {
512         "sentence_id":73,
513         "emotion":"agreeable",
514         "emotion_direction":1,
515         "confidence_emotion":0.92,
516         "intensity_emotion":0.75,
517         "personality_trait":"accommodating",
518         "confidence_trait":0.85
519     }
520     {
521         "sentence_id":75,
522         "emotion":"efficient",
523         "emotion_direction":1,
524         "confidence_emotion":0.93,
525         "intensity_emotion":0.88,
526         "personality_trait":"professional",
527         "confidence_trait":0.9
528     }
529     {
530         "sentence_id":76,
531         "emotion":"helpful",
532         "emotion_direction":1,
533         "confidence_emotion":0.95,
534         "intensity_emotion":0.9,
535         "personality_trait":"customer service oriented",
536         "confidence_trait":0.92
537     }
538     {
539         "sentence_id":78,
540         "emotion":"thankful",
541         "emotion_direction":1,
542         "confidence_emotion":0.93,
543         "intensity_emotion":0.8,
544         "personality_trait":"polite",
545         "confidence_trait":0.88
546     }
547     {
548         "sentence_id":81,
549         "emotion":"appreciative",
550         "emotion_direction":1,
551         "confidence_emotion":0.92,
552         "intensity_emotion":0.75,
553         "personality_trait":"friendly",
554         "confidence_trait":0.9
555     }
556 ]
557 }
558 "transcript":[
559     {
560         "speaker":"Speaker0",
561         "text":"H: Good afternoon",
562         "startTime":1.87,
563         "endTime":1.89,
564         "sentence_id":0
565     }
566     {
567         "speaker":"Speaker0",
568         "text":"thank you for calling voice and support",
569         "startTime":3,
570         "endTime":5.2,
571         "sentence_id":1
572     }
573     {
574         "speaker":"Speaker0",
575         "text":"my name is Steve",

```

```
576     "startTime":6.16,
577     "endTime":7.2,
578     "sentence_id":2
579 }
580 {
581     "speaker":"Speaker0",
582     "text":"how can i assist you today",
583     "startTime":8,
584     "endTime":9.64,
585     "sentence_id":3
586 }
587 {
588     "speaker":"Speaker1",
589     "text":"hi i'm having",
590     "startTime":13.46,
591     "endTime":13.48,
592     "sentence_id":4
593 }
594 {
595     "speaker":"Speaker1",
596     "text":"some issues with clash phone service",
597     "startTime":14.88,
598     "endTime":16.6,
599     "sentence_id":5
600 }
601 {
602     "speaker":"Speaker1",
603     "text":"my calls keep dropping and the quality is really bad especially when I'm talking to peoplere",
604     "startTime":17.6,
605     "endTime":23.56,
606     "sentence_id":6
607 }
608 {
609     "speaker":"Speaker0",
610     "text":"I'm sorry to hear that you are experiencing this issue",
611     "startTime":26.83,
612     "endTime":29.76,
613     "sentence_id":8
614 }
615 {
616     "speaker":"Speaker0",
617     "text":"I'll do my best to help you resolve them can i start by getting your account number or the phone",
618     "startTime":31.12,
619     "endTime":38.28,
620     "sentence_id":9
621 }
622 {
623     "speaker":"Speaker0",
624     "text":"with your account",
625     "startTime":39.12,
626     "endTime":39.64,
627     "sentence_id":10
628 }
629 {
630     "speaker":"Speaker1",
631     "text":"my phone number is",
632     "startTime":44.72,
633     "endTime":46.32,
634     "sentence_id":11
635 }
636 {
637     "speaker":"Speaker1",
638     "text":"0727672",
639     "startTime":47.32,
640     "endTime":47.32,
641     "sentence_id":12
642 }
```

```
643 {
644   "speaker": "Speaker1",
645   "text": "727 672",
646   "startTime": 48.2,
647   "endTime": 50.2,
648   "sentence_id": 13
649 }
650 {
651   "speaker": "Speaker1",
652   "text": "it",
653   "startTime": 52.4,
654   "endTime": 52.4,
655   "sentence_id": 14
656 }
657 {
658   "speaker": "Speaker0",
659   "text": "thank you",
660   "startTime": 54.56,
661   "endTime": 55.36,
662   "sentence_id": 15
663 }
664 {
665   "speaker": "Speaker0",
666   "text": "let me pull up your account",
667   "startTime": 56.52,
668   "endTime": 58.2,
669   "sentence_id": 16
670 }
671 {
672   "speaker": "Speaker0",
673   "text": "one moment please",
674   "startTime": 59.36,
675   "endTime": 60.48,
676   "sentence_id": 17
677 }
678 {
679   "speaker": "Speaker0",
680   "text": "okay i've got your account here can you can you tell me when you started experiencing",
681   "startTime": 64.64,
682   "endTime": 70.88,
683   "sentence_id": 19
684 }
685 {
686   "speaker": "Speaker0",
687   "text": "this issue sue",
688   "startTime": 71.76,
689   "endTime": 72.24,
690   "sentence_id": 20
691 }
692 {
693   "speaker": "Speaker1",
694   "text": "started a few days ago at first it was just a couple of calls dropping but now it's happening mo",
695   "startTime": 72.24,
696   "endTime": 84,
697   "sentence_id": 21
698 }
699 {
700   "speaker": "Speaker1",
701   "text": "and the call quality has been really bad",
702   "startTime": 84.84,
703   "endTime": 87.56,
704   "sentence_id": 23
705 }
706 {
707   "speaker": "Speaker0",
708   "text": "have you noticed if this happened in specific location or if it's",
709   "startTime": 92.64,
```

```
710     "endTime":97.01,
711     "sentence_id":25
712 }
713 {
714     "speaker":"Speaker0",
715     "text":"happening everywhere you go",
716     "startTime":98.2,
717     "endTime":99.96,
718     "sentence_id":26
719 }
720 {
721     "speaker":"Speaker1",
722     "text":"it is mostly",
723     "startTime":103.31,
724     "endTime":103.33,
725     "sentence_id":27
726 }
727 {
728     "speaker":"Speaker1",
729     "text":"happening at home but I've noticed it a few times when I'm out as well",
730     "startTime":104.72,
731     "endTime":109.56,
732     "sentence_id":28
733 }
734 {
735     "speaker":"Speaker0",
736     "text":"thank you for that information",
737     "startTime":111.04,
738     "endTime":112.64,
739     "sentence_id":29
740 }
741 {
742     "speaker":"Speaker0",
743     "text":"and to better understand the issue can you let me know can you please let me know if you're using",
744     "startTime":113.8,
745     "endTime":123.24,
746     "sentence_id":30
747 }
748 {
749     "speaker":"Speaker1",
750     "text":"I'm using",
751     "startTime":125.07,
752     "endTime":125.08,
753     "sentence_id":31
754 }
755 {
756     "speaker":"Speaker1",
757     "text":"an iPhone",
758     "startTime":126.56,
759     "endTime":127.12,
760     "sentence_id":32
761 }
762 {
763     "speaker":"Speaker0",
764     "text":"great thank you let's try a few troubleshooting",
765     "startTime":131.36,
766     "endTime":134.56,
767     "sentence_id":33
768 }
769 {
770     "speaker":"Speaker0",
771     "text":"steps to see if we can improve your service",
772     "startTime":135.48,
773     "endTime":137.92,
774     "sentence_id":34
775 }
776 {
```

```
777     "speaker": "Speaker0",
778     "text": "and first",
779     "startTime": 139.04,
780     "endTime": 139.68,
781     "sentence_id": 35
782 }
783 {
784     "speaker": "Speaker0",
785     "text": "have you tried restarting your phone since this issue app began app began",
786     "startTime": 140.72,
787     "endTime": 144.74,
788     "sentence_id": 37
789 }
790 {
791     "speaker": "Speaker1",
792     "text": "12 yes",
793     "startTime": 128.16,
794     "endTime": 147.52,
795     "sentence_id": 38
796 }
797 {
798     "speaker": "Speaker1",
799     "text": "I've tried",
800     "startTime": 148.48,
801     "endTime": 149.12,
802     "sentence_id": 39
803 }
804 {
805     "speaker": "Speaker1",
806     "text": "studying it a few times but it didn't seem to help.",
807     "startTime": 150.32,
808     "endTime": 153.4,
809     "sentence_id": 40
810 }
811 {
812     "speaker": "Speaker0",
813     "text": "understood and let's check your network",
814     "startTime": 155.52,
815     "endTime": 159.44,
816     "sentence_id": 41
817 }
818 {
819     "speaker": "Speaker0",
820     "text": "setting next can you go to please",
821     "startTime": 160.24,
822     "endTime": 162.72,
823     "sentence_id": 42
824 }
825 {
826     "speaker": "Speaker0",
827     "text": "settings then",
828     "startTime": 163.56,
829     "endTime": 164.88,
830     "sentence_id": 43
831 }
832 {
833     "speaker": "Speaker0",
834     "text": "cellular and there can you please make sure cellular data is turned on also",
835     "startTime": 165.88,
836     "endTime": 172.48,
837     "sentence_id": 45
838 }
839 {
840     "speaker": "Speaker0",
841     "text": "check if you have a voice and data set to 4G or LTE",
842     "startTime": 173.32,
843     "endTime": 177.92,
```



```
844     "sentence_id":46
845 }
846 {
847     "speaker":"Speaker1",
848     "text":"and i check",
849     "startTime":182.56,
850     "endTime":183.76,
851     "sentence_id":47
852 }
853 {
854     "speaker":"Speaker1",
855     "text":"and cellular data is on and it's set to LTE",
856     "startTime":185.08,
857     "endTime":188.28,
858     "sentence_id":48
859 }
860 {
861     "speaker":"Speaker0",
862     "text":"thanks for confirming",
863     "startTime":190.56,
864     "endTime":192.12,
865     "sentence_id":49
866 }
867 {
868     "speaker":"Speaker0",
869     "text":"and now let's",
870     "startTime":193.36,
871     "endTime":194.4,
872     "sentence_id":50
873 }
874 {
875     "speaker":"Speaker0",
876     "text":"try resetting your network setting this won't erase any personal data but we'll reset wi-fi blue
877     "startTime":195.96,
878     "endTime":203.08,
879     "sentence_id":51
880 }
881 {
882     "speaker":"Speaker0",
883     "text":"to do this go to settings",
884     "startTime":204.2,
885     "endTime":206.08,
886     "sentence_id":52
887 }
888 {
889     "speaker":"Speaker0",
890     "text":"then general",
891     "startTime":207.04,
892     "endTime":207.8,
893     "sentence_id":53
894 }
895 {
896     "speaker":"Speaker0",
897     "text":"then reset",
898     "startTime":208.8,
899     "endTime":209.4,
900     "sentence_id":54
901 }
902 {
903     "speaker":"Speaker0",
904     "text":"and there you have recent reset network settings",
905     "startTime":210.28,
906     "endTime":214.04,
907     "sentence_id":56
908 }
909 {
910     "speaker":"Speaker0",
```

```
911     "text": "please let me know once you have done that",
912     "startTime": 215.04,
913     "endTime": 217.4,
914     "sentence_id": 57
915 }
916 {
917     "speaker": "Speaker1",
918     "text": "all right",
919     "startTime": 221.14,
920     "endTime": 221.15,
921     "sentence_id": 58
922 }
923 {
924     "speaker": "Speaker1",
925     "text": "i reset the network setting",
926     "startTime": 222.96,
927     "endTime": 224.8,
928     "sentence_id": 59
929 }
930 {
931     "speaker": "Speaker1",
932     "text": "just now",
933     "startTime": 225.84,
934     "endTime": 226.32,
935     "sentence_id": 60
936 }
937 {
938     "speaker": "Speaker0",
939     "text": "now let's test your phone and try making a call and see if this issue persists i'll stay on the",
940     "startTime": 229.88,
941     "endTime": 239.4,
942     "sentence_id": 62
943 }
944 {
945     "speaker": "Speaker1",
946     "text": "making a call right now please let me check",
947     "startTime": 244.08,
948     "endTime": 246.32,
949     "sentence_id": 63
950 }
951 {
952     "speaker": "Speaker1",
953     "text": "and the call quality seems to be a bit better but I'm not sure if it's completely fix you know i",
954     "startTime": 254.96,
955     "endTime": 264.93,
956     "sentence_id": 65
957 }
958 {
959     "speaker": "Speaker0",
960     "text": "first of all i'm glad to hear there is some improvement",
961     "startTime": 268,
962     "endTime": 270.68,
963     "sentence_id": 66
964 }
965 {
966     "speaker": "Speaker0",
967     "text": "sometimes this issue can be due to a temporary network congestion or interference",
968     "startTime": 272.36,
969     "endTime": 276.72,
970     "sentence_id": 67
971 }
972 {
973     "speaker": "Speaker0",
974     "text": "if this problem continues",
975     "startTime": 278.16,
976     "endTime": 279.36,
977     "sentence_id": 68
```

```
978     }
979     {
980         "speaker": "Speaker0",
981         "text": "we can look into it further",
982         "startTime": 280.36,
983         "endTime": 281.92,
984         "sentence_id": 69
985     }
986     {
987         "speaker": "Speaker0",
988         "text": "would you like me to create a ticket for our technical team to investigate",
989         "startTime": 283.04,
990         "endTime": 286.76,
991         "sentence_id": 70
992     }
993     {
994         "speaker": "Speaker1",
995         "text": "that will be great I really need this result as soon as possible",
996         "startTime": 290.24,
997         "endTime": 294.08,
998         "sentence_id": 72
999     }
1000    {
1001        "speaker": "Speaker0",
1002        "text": "of course of course",
1003        "startTime": 295.76,
1004        "endTime": 296.56,
1005        "sentence_id": 73
1006    }
1007    {
1008        "speaker": "Speaker0",
1009        "text": "i'll create a ticket for you right now right now one moment",
1010        "startTime": 297.44,
1011        "endTime": 302,
1012        "sentence_id": 75
1013    }
1014    {
1015        "speaker": "Speaker0",
1016        "text": "and you will receive an update within the next 24 to 48 hours is there anything else i can assist",
1017        "startTime": 303.12,
1018        "endTime": 310.8,
1019        "sentence_id": 76
1020    }
1021    {
1022        "speaker": "Speaker1",
1023        "text": "No that's",
1024        "startTime": 313.12,
1025        "endTime": 313.13,
1026        "sentence_id": 77
1027    }
1028    {
1029        "speaker": "Speaker1",
1030        "text": "all for now thank you for your help",
1031        "startTime": 314.16,
1032        "endTime": 316.36,
1033        "sentence_id": 78
1034    }
1035    {
1036        "speaker": "Speaker0",
1037        "text": "you're welcome thank you for your patience and being a valued customer have a great day",
1038        "startTime": 318.67,
1039        "endTime": 324.36,
1040        "sentence_id": 81
1041    }
1042    {
1043        "speaker": "Speaker1",
1044        "text": "you thank you goodbye.",
```

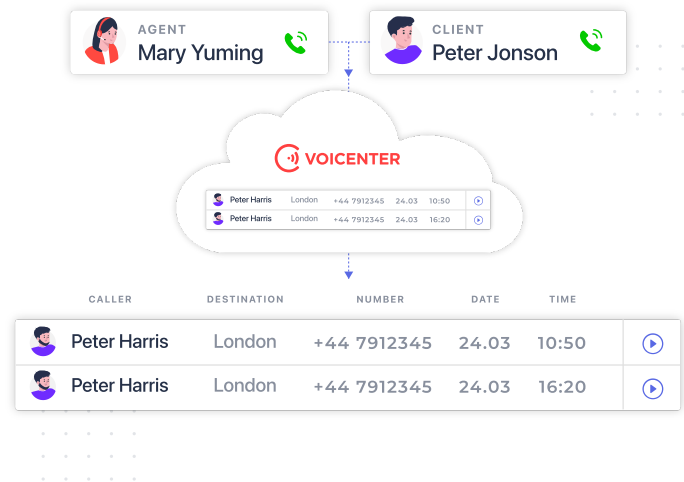
```
1045         "startTime":326.4,
1046         "endTime":null,
1047         "sentence_id":82
1048     }
1049     {
1050         "speaker":"Speaker0",
1051         "text":"goodbye",
1052         "startTime":330.28,
1053         "endTime":null,
1054         "sentence_id":83
1055     }
1056 ]
1057 }
1058 "DepartmentID":1234567,
1059 "DepartmentName":"Voicenter",
1060 "TopDepartmentID":1234567,
1061 "TopDepartmentName":"Voicenter"
1062 }
```

# Test Json

---

# Call Log API

Call log API allows you to pull call record details (CDR) by a simple web request. You can receive up to 10,000 CDR result per request by a filter of your choice. You can analyze the data and generate automated reports regarding your call center performance, store the call history of your clients along with the url for the call record.



## The Call-Log API works as follows:

The first part of the Call-Log api allows you the set which call records details (CDR's) you will receive based on your filter (search criteria) and your request.

You will decide which on which search criteria to use in order to receive your CDR's, the search criteria will based on:

From date & to date

Phone numbers

extensions

IdentityCriteria

CallID

cdrTypes.

Secondly, you decide which CDR fields will be returned to you by Voicenter.

You decide what information you will receive in your request to Voicenter.

You can set to get only certain CDR fields like record url, status and type of the call, rest of the CDR fields will not be returned.

For security purposes you need to send your request with a code to authenticate your account and make sure the IP address of your server which will be sending api request to Voicenter is authorized in your account.

The code can be provided by our Backoffice team.

## General Description

### API URI

1 | <https://api.voicenter.com/hub/cdr/>

## Acceptable Request Types

POST

GET

## Possible Response Formats

JSON

## Request Parameters

As stated above, the Call Log api is divided into 2 "parts":  
The first part you need to authenticate and set the filter (search criteria) which will decide what CDR's you will receive.  
The second part you will need to set which fields that hold the call details record will be returned to you by Voicenter.

## Search Criteria Options

| Parameter        | Description   | Type  |
|------------------|---|---|
| code             | String. A request can also be send with bearer token.   | A unique organization identifier.<br>Mandatory. |
| fromdate         | Date time in ISO 8601 format.<br>Important – the value is in GMT 0.   | Mandatory.                                      |
| todate           | Date time in ISO 8601 format.<br>Important – the value is in GMT 0.   | Mandatory.                                      |
| phones           | Array of strings.<br>Each string is phone number to search include country code.<br>Israel prefix can be filter without the country code. | Optional.                                       |
| extensions       | Array of strings. Each string is Voicenter SIP extension.   | Optional.                                       |
| IdentityCriteria | One of the following: Account, Hierarchical, Department, User.  | Optional.                                       |
| callID           | String.<br>Filter the search for a specific Voicenter call id.  | Optional.                                       |
| cdrTypes         | Array of integers.<br>Allows to filter CDRs by a specific call types (incoming, click2call,dialer etc.).<br>E.g. [4, 18].                 | Optional.                                       |
| campaignID       | Array of numbers, each number is a Voicenter campaign ID.   | Optional.                                       |
| queueID          | Array of numbers, each number is a Voicenter queue ID.  | Optional.                                       |

## Sort

You can decide how to sort the information in the response you receive from Voicenter.

| Parameter | Description  | Type |
|-----------|--|------|
| field     | One of the fields list which the sorting criteria will take place. |      |
|           |  |      |

| Parameter | Description  | Type |
|-----------|--|------|
| order     | asc \ desc.<br>which order to organize the returned results – ascending or descending. |      |

\* It is an Array type, you can send requests with multiple field & order.

## Requeust Parameters

These are the CDR fields you can configure to receive in the Response from Voicenter.

Below is an explanation about each field.

| Field        | Description   | Example   |
|--------------|---|---|
| CallerNumber | Displays the caller's phone number.<br>The phone number that will show to the call destination.   | "CallerNumber": "0722776772"  |
| TargetNumber | Displays the destination of the call.<br>Can be a phone number or the extension SIP code.<br>The phone number value will be sent with the international country prefix.   | "TargetNumber": "AAPINFzL" /<br>"TargetNumber": "972722776772"                    |
| Date         | Displays the time and date that the call was made.  | "Date": "2022-007-31 09:59:29"  |
| DateEpoch    | Displays the time and date that the call was made in Epoch format.  | "DateEpoch": 1554027973   |
| Duration     | Displays the duration of the call (seconds). This duration does not include the ringing duration, only the actual time of the conversation that was made.   | "Duration": 33  |
| CallID       | Displays the ID code of the specific call.  | "CallID": "20200720095ilila137b2f69s"   |
| Type         | Displays the type of Call.<br>For example: if it is an incoming/outgoing call?<br>There are several call types (More details are down below).   | "Type": "Incoming Call" / "Type": "Extension Outgoing" / "Type": "Click2Callleg1" |
| CdrType      | The "Type" field returns the call type.<br>The "CdrTypes" field returns the corresponding numeric value of the call type.<br>The digit id of each call type can be seen below at the Call Type table.   | "TargetPrefixName": 4   |
| DialStatus   | Displays what happened with the specific call?<br>If the call had 2 legs, this field will show the status of the leg1 part of the call (call types: Click2call, Automatic Dialer calls).<br>There are several call statuses (More details are in the next pages). | "DialStatus": "ANSWER" /<br>"DialStatus": "ABANDONE" /<br>"DialStatus": "TE"      |



| Field              | Description  | Example   |
|--------------------|--|---|
| Targetextension    | <p>Displays the extension SIP code that received the incoming call.</p> <p>Sometimes this value will be identical to the "TargetNumber" field.</p> <p>There are cases that the incoming call is received not directly to the extension, so the "TargetNumber" can display different value.</p> | "Targetextension":"AAPINFzL"  |
| Callerextension    | <p>Displays the extension SIP code that the call was dialed from.</p> <p>This value is different from the "CallerNumber" field.</p> <p>In the "CallerNumber" field it displays the actual number that will appear to the call destination.</p>   | "Callerextension":"AAPINFzL"  |
| DID                | <p>Displays the origin phone number that the caller called to.</p> <p>This field will only show value if the call direction was an incoming call.</p>  | "DID":"0722776772"  |
| QueueName          | If the call was directed to a queue service, it displays the queue name.   | "QueueName":"Service Queue"   |
| RecordURL          | Displays a URL link to the call recording.   | "RecordURL":"https://cpanel.voicenter.co.il/CallsHistory/PlayRecord/123abc.mp3" |
| RecordExpect       | <p>A Boolean type field.</p> <p>Displays if the call was recorded or not.</p>  | "RecordExpect":true   |
| Price              | Displays the total price of the call in ILS cents (Agorot).  | "Price":7   |
| RingTime           | <p>Displays the ringing duration of the call(seconds).</p> <p>Not include the actual conversation duration.</p>  | "RingTime":23   |
| RepresentativeName | <p>Displays the Voicenter representative name that the specific call was associated with.</p> <p>This field will only be displayed if the user was logged in to an extension when the call was made.</p>   | "RepresentativeName":"Walter Melon"   |
| RepresentativeCode | <p>Displays the Voicenter representative ID code that the specific call was associated with.</p> <p>This field will only be displayed if the user was logged in to an extension when the call was made.</p>  | "RepresentativeCode":"19996"  |
|                    |  |   |

| Field            | Description   | Example   |
|------------------|---|---|
| UserName         | Similar to the field "RepresentativeName".<br>If a user was logged in to an extension when the call was made, this field will show the same value as "RepresentativeName".<br>But if a user was not logged in then this field will show the user name that the extension\DID is associated with.  | "UserName":"Walter Melon"   |
| DTMFData         | A JSON array type field. Displays the stages in the IVR that an incoming call went through.<br>Fields:<br>"LayerName" – The IVR layer name as it appears in the Voicenter control panel (CPanel).<br>"DTMF" – If the caller pressed on a digit\s. If the caller did not press an any digit the default value will be "0".<br>"LayerNumber" - The IVR layer ID.  | "DTMFData": "[{LayerName: \"Main Menu\", DTMF: 2, LayerNumber: \"0\"}, {LayerName: \"Sales\", DTMF: 0, LayerNumber: \"15\"}]"                                     |
| CustomData       | A JSON array type field. Displays custom data that was sent to us.<br>In addition, if a specific call has an origin call it will display here with the field name "OriginalIvrUniqueID".<br>For example: if a call was first answered by one representative who then transferred it to another representative.<br>The second part of the call that was transferred, will have in the "CustomData" field a callID value of the original call, thus associating both calls. | "CustomData":<br>{\"OriginalIvrUniqueID\":\"2020072118102as sdsdas\"} / \"CustomData\":<br>{\"var_clientID\":\"2118102\",<br>\"var_clientAccountID\":\"123456\"}" |
| DepartmentName   | Displays the department\account name (as it appears in Voicenter) that the call was associated with.  | "DepartmentName":"Voicenter Account"  |
| DepartmentId     | Displays the department\account ID (as it appears in Voicenter) that the call was associated with.  | "DepartmentId":123456   |
| TargetPrefixName | Displays the name of the country that the call was dialed to.   | "TargetPrefixName":"Israel"   |

## Call Type

In Voicenter we have different call type as detailed in the table below.

You can combine the call type and call statuses parameters to generate automated processes in your organization.

For example, if your sales tried to contact a lead via Click2call of an outgoing call (call types) and failed to receive an answer (call status) for the potential client you can send an SMS/WhatsApp/email to that lead informing them you tried to contact them.

| CDR type name          | CDR type id | Description  |
|------------------------|-------------|--|
| Incoming Call          | 1           | A regular incoming call that was dialed to a phone number and ended up in the IVR or an extension (did not end in a queue, which will make it type 4).   |
| CC                     | 2           | A call that was made through a calling card (Access number) service.   |
| Extension Outgoing     | 4           | A regular outgoing call (manually dialed from the phone).  |
| Queue                  | 8           | An incoming call that was received by a queue.   |
| Click2Call leg1        | 9           | A call that was made by click2call (Not by Dialer).<br>Leg 1 of the call.<br>Leg 1 - the initial connection of the call to the extension.<br>This is the "Phone" parameter in the Click2call API.  |
| Click2Call leg2        | 10          | A call that was made by click2call (Not by Dialer).<br>Leg 2 of the call.<br>Leg 2 - the actual call that is being made to the destination. This is the "Target" parameter in the Click2call API.  |
| VoiceMail              | 11          | A call that was answered by Voicenter voicemail.   |
| Callference            | 12          | A call that was made through Voicenter callference service.  |
| XferCDR                | 13          | A call that manually transferred from an extension to a DID/another extension.   |
| ProductiveCall Leg1    | 14          | A "leg 1" Agents Auto Dialer calls. Leg1 - the initial connection of the call to the extension.  |
| ProductiveCall Leg2    | 15          | A "leg 2" Agents Auto Dialer calls. Leg 2 -the actual call that is being made to the destination.  |
| Scrubber               | 16          | A call that was made through Voicenter's Scrubber service.   |
| Click 2 IVR            | 17          | "Leg1" Predictive Dialer calls.<br>Leg1 - the initial connection of the call to the destination.   |
| Click 2 IVR Incoming   | 18          | This is the "Leg 2" Predictive Dialer call.<br>After the initial Leg1 (Click 2 IVR) call was answered, this new leg is dialing the IVR (a layer in Voicenter telephony menu) and Leg 1 is connected to the representatives.<br>This call type is for all Click 2 IVR that did not enter a queue.<br>If the call enters a queue service the type will be type 15. |
| Click 2 Queue Incoming | 19          | This is the "Leg 2" Predictive Dialer call.<br>After the initial Leg1 (Click 2 IVR) call was answered, this new leg is dialing the IVR (a layer in Voicenter telephony menu) and Leg 1 is connected to the representatives.<br>If the call enters a queue, then the call status will be - Click 2 Queue. Else, the call type will be type 14.                    |
| FaxCdr                 | 20          | A call that was made through Voicenter's internal outgoing fax service.<br>*Not released yet.  |
|                        |             |  |

| CDR type name     | CDR type id | Description  |
|-------------------|-------------|--|
| Attended CDR leg1 | 21          | A call that was transferred with consultation.   |
| Attended CDR leg2 | 22          | A call that was transferred with consultation.<br>This type will only be made in a case of an incoming call that was answered by representative "A", "A" will then put the call on hold and make another call(consult) to another person - "B". Afterward, "A" will transfer the initial call to "B".<br>The "Leg 2" is the part of the call between "B" and the initial caller. |
| Auto forward      | 23          | A call that was automatically transferred from an extension to a DID (usually representatives configure their phones manually to transfer calls when they are not available).  |

## Call statuses

In Voicenter we have different call statuses as detailed below.

Along with the call type (provided above) you can create an automated process, for example send an email to your call center manager asking to callback a client who abandoned (call status) the queue (call type) or tried to join when the queue was full (call status) with other callers.

You can use the information provided by our Call Log api to perform many automated process your organizational business logic dictates.

| CDR Status Name | Description  |
|-----------------|--|
| NOTDIALED       | Hang-up occurred before the call was made.   |
| ANSWER          | A call was answered. A successful dial.<br>A successful dial.<br>The caller reached the destination.<br>Whenever we receive an answer response signal, also when the call reached local voicemail service and etc. |
| BUSY            | Busy signal.<br>The dial command reached its number but the number is busy.  |
| NOANSWER        | No answer.<br>The dial command reached its number, the number rang for too long, then the dial timed out.  |
| CANCEL          | A call is canceled.<br>The dial command reached its number but the caller hung up before the call destination could answer.  |
| ABANDONE        | When using Voicenter's queue service, the caller hung up before the call destination could answer.   |
| TIMEOUT         | When using Voicenter's queue service - a call has reached timeout in the queue.  |
| FULL            | When using Voicenter's queue service - a call tried to enter a queue service, but the queue reached the maximum amount of waiting callers.   |

|  |
|--|
|  |
|--|

| CDR Status Name | Description   |
|-----------------|---|
| EXIT            | When using Voicenter's queue service - a caller while waiting in the queue, chose to exit from the queue.                                   |
| JOINEMPTY       | When using Voicenter's queue service - a call tried to enter a queue service, but there were no representatives who were logged-in to it.   |
| VOEND           | Hang-up during IVR without actual dialing.<br>In this case, the caller waited in the IVR but hung up before the call rang in any extension. |
| TE              | When an incoming call is directed to an IVR recording and afterward it configured to hung up the call.                                      |
| NOTCALLED       | A Leg2 Click2Call was not called.<br>When using a click2call service and the Leg1 initial stage was not successful (not answered).          |
| VOICEMAIL       | Call entered to Voicenter voicemail service.  |

## Call Error Types

These error responses display cases when there was a problem connecting to the target destinations. It mainly used for Voicenter internal Tracking.

| Error Types | Description  |
|-------------|--|
| CONGESTION  | Congestion.<br>This status is usually a sign that the dialed number is not recognized. |
| CHANUNAVAIL | Channel unavailable.<br>On SIP, peer may not be registered.                            |
| INVALIDARGS | Error parsing dial command arguments.  |
| SSWPREAUTH  | SSW outgoing call cancel before actual dial.   |

## Get Reqeust Example

Here below is an example for a reqeust to Call Log API in a POST JSON format. You can see the fields which you want to receive back in the response. You can see the search criteria you can use and sorting order.

```
1 | https://api.voicenter.com/hub/cdr/?code=xxxxxxxxxxxxxxxxxxx&fromdate=2021-10-02T08:09:40&todate=2021-11-
2 | 09T23:59:40&phones=972501234567&extensions=SipSip&fields=Date&fields=Type&Fields=DID&Fields=CallerNumber
```

## JSON Request Example

Here below is an example for a reqeust to Call Log API in a POST JSON format. You can see the fields which you want to receive back in the response. You can see the search criteria you can use and sorting order.

```
1 | {
```

```
2  "code": "xxxxxxxxxxxxxxxxxxxxxxxxxxxx",
3  "fields": [
4      "CallerNumber",
5      "TargetNumber",
6      "Date",
7      "DateEpoch",
8      "Duration",
9      "CallID",
10     "CustomData",
11     "Type",
12     "CdrType",
13     "DialStatus",
14     "TargetExtension",
15     "CallerExtension",
16     "DID",
17     "QueueName",
18     "RecordURL",
19     "Price",
20     "RingTime",
21     "RepresentativeName",
22     "RepresentativeCode",
23     "DTMFData",
24     "DepartmentName",
25     "DepartmentId"
26 ]
27 "search": {
28     "fromdate": "2021-10-02T08:09:40",
29     "todate": "2021-11-09T23:59:40",
30     "cdrTypes": [
31         1,
32         8,
33         9
34     ]
35     "phones": [
36         "972501234567",
37         "0722776772"
38     ]
39     "extensions": [
40         "SipSip"
41     ]
42     "IdentityCriteria": "Account",
43     "CallID": "xxxxxxxxxxxxxxxxxxxxxxxx"
44 }
45 "sort": [
46     {
47         "order": "desc",
48         "field": "date"
49     }
50     {
51         "order": "asc",
52         "field": "Duration"
53     }
54 ]
55 }
```

## Response Parameters

Below are the parameters which are included in the response

| Parameter | Description |
|-----------|-------------|
|           |             |

| Parameter         | Description  |
|-------------------|--|
| ERROR_NUMBER      | Integer.<br>Represents error number.   |
| ERROR_DESCRIPTION | String.<br>Represents error description.   |
| STATUS_CODE       | Integer.<br>Http response code.  |
| TOTAL_HITS        | Integer.<br>The total sum of CDRs results of the client request.                                     |
| RETURN_HITS       | Integer.<br>How many CDRs results returned in the response(we limit each request up to 10,000 CDRs). |
| CDR_LIST          | Array of objects.<br>Each object is call detailed record, as requested.                              |

## JSON Response Example

After sending a request to Voicenter, you will receive a response which will enable you to receive the CDR's according to your search criteria.  
Below is an example of a response you will receive from Voicenter.

```
1  {
2    "ERROR_NUMBER":0,
3    "ERROR_DESCRIPTION":"OK",
4    "STATUS_CODE":200,
5    "TOTAL_HITS":2,
6    "RETURN_HITS":2,
7    "CDR_LIST":[
8      {
9        "CallerNumber":"0722776772",
10       "TargetNumber":"972501234567",
11       "Date":"2021-10-10T09:04:58Z",
12       "Duration":2,
13       "CallID":"202110101204550233243ghff3189e5c",
14       "CustomData":{"
15         }
16       "Type":"Extension Outgoing",
17       "CdrType":4,
18       "DialStatus":"ANSWER",
19       "TargetExtension":"",
20       "CallerExtension":"SipSip",
21       "DID":"",
22       "RecordURL":"https://cpanel.voicenter.co.il/CallsHistory/PlayRecord/2021101043950926.mp3",
23       "RingTime":10,
24       "RepresentativeName":"rep 1",
25       "RepresentativeCode":"87654321",
26       "DTMFData":[
27         ]
28       "DepartmentName":"Test Account",
29       "DepartmentId":12345678
30     }
31     {
32       "CallerNumber":"0501234567",
33       "TargetNumber":"SipSip",
```

```

34     "Date": "2021-10-02T12:55:18Z",
35     "Duration": 0,
36     "CallID": "202110021555160180fd3jhj455",
37     "CustomData": {
38         "OriginalIvrUniqueID": "20211002156fdf534hfg453df34"
39     }
40     "Type": "Incoming Call",
41     "CdrType": 1,
42     "DialStatus": "CHANUNAVAIL",
43     "TargetExtension": "SipSip",
44     "CallerExtension": "",
45     "DID": "0722776772",
46     "RecordURL": "",
47     "RingTime": 2,
48     "RepresentativeName": "rep 1",
49     "RepresentativeCode": "87654321",
50     "DTMFData": [
51         {
52             "LayerName": "rep 1",
53             "DTMF": 0,
54             "LayerNumber": "1"
55         }
56     ]
57     "DepartmentName": "Test Account",
58     "DepartmentId": 12345678,
59     "TargetPrefixName": ""
60 }
61 ]
62 }

```

## Error Number Response Description

Upon receiving the response from Voicenter there will be an error number and the description of the error. Below is a detailed explanation.

| ERROR_NUMBER | ERROR_DESCRIPTION                                  | STATUS_CODE | Description   |
|--------------|--|-------------|---|
| 0            | OK   | 200         | The request was sent successfully.  |
| 1            | Request limit exceeded.<br>Please try again later. | 403         | There is a limit to how many requests a client can send in a specific period of time.<br>It is recommended to send one request every 5 seconds. |
| 2            | Authorization failed.                              | 403         | Code value is not valid.  |
| 4            | IP address xx.xx.xx.xx is not trusted.             | 403         | The request is sent from unauthorized IP address.   |
| 5            | Out of range                                       | 404         | There was a problem with search fields: fromdate, todater.<br>The search range is invalid.  |

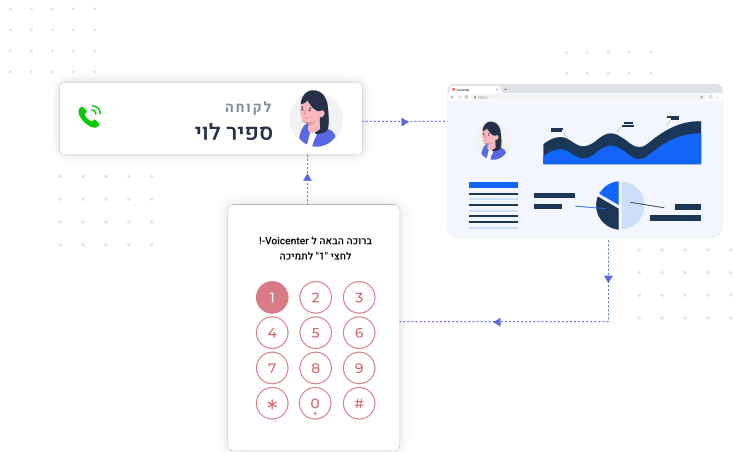
## Service Limits and security



- The CDRs are updated in the service a few minutes after the calls are ended.
- Can receive up to 10,000 CDR results per request.
- Can send up to 30 requests per one minute.
- Requests can only be sent from an authorized IP address that can be set in the Cpanel.

# External Layer API

Extend the functionality of your Voicenter IVR and implement call routing based on the information in your organizational database or CRM systems in order to perform advanced routing of incoming calls.



## GENERAL DESCRIPTION

### Voicenter Request

With External IVR API, you can extend functionality of Voicenter IVR and connect it to your organizational database or CRM systems in order to perform advanced routing of incoming calls. For example, you can identify incoming call by phone number and determine if this is an existing client or a new potential client, and perform different routing scenarios accordingly, such as:

Caller number is registered as an existing lead: Your CRM can check if one of your sales representatives already spoke to this client and reply back with the layer of this representative in order to redirect the call directly to that specific sales representative. You want to do that in order to finalize sales process in a smoother more customer experience oriented IVR.

Caller number is registered as an existing account: Your CRM can check if there is an ongoing open ticket/case and transfer the call to the right customer services representative/desk - in order to resolve the issue and provide personal service faster.

Caller number does not exist in your CRM: Your CRM can transfer these calls to sales representatives or as an alternative you can ask from the client to provide some identification information in order to transfer the call to the right desk.

In addition, callers can be asked to dial their personal ID, account number or anything else, and this data can be process by your organizational database or CRM systems in order to perform varied actions. These digits pressed during the navigation in the IVR are called DTMFs.

External IVR API utilizes existing layers of Voicenter IVR in order to route calls. Further explanation about creating and managing IVR layer can be found at the end of this chapter.

In order to utilize advanced features of the External Layer IVR, the system needs an end point URL to address when calling the Layer. This URL can be managed in the Layer settings tab.

Once the call is received, Voicenter IVR will initiate a POST request to this endpoint. This request will contain the following information:

Reqeust Format

1. POST-JSON

Response Format

1. POST-JSON

Request Parameters

After implementing your endpoint in the relevant layer/s in your Voicenter IVR you will start receiving request with the following parameters:

| Field Name        | Description   | Type    |
|-------------------|---|---------|
| METHOD            | Will always be "IVR_LAYER_INPUT".   | String  |
| DATA              | Data object of an IVR request.  | String  |
| DID               | DID number that the call was received on(Incoming number).  | String  |
| CALLER_ID         | Caller ID of the current caller.  | String  |
| IVR_UNIQUE_ID     | Voicenter unique ID of a call. Can be used for state purposes.  | String  |
| DTMF              | If the caller was requested to enter identification information, this input will be passed in this parameter.<br>*In case the caller did not enter any value(did not make any dial action), the default value will always be “0”. | String  |
| LAYER_ID          | The current layer ID in the IVR that the request is sent from.  | Integer |
| PREVIOUS_LAYER_ID | If the call was passed to the layer from another layer, the previous layer will be shown here.  | Integer |

Reqeust Example

Below is an example of a request your endpoint will receive from the Voicenter IVR.

```
1  {
2    "METHOD": "IVR_LAYER_INPUT",
3    "DATA": {
4      "DID": "0722776772",
5      "CALLER_ID": "0501234567",
6      "IVR_UNIQUE_ID": "1bcd7954224861f85a2d70612f2",
7      "DTMF": "1234",
8      "LAYER_ID": "5",
9      "PREVIOUS_LAYER_ID": "5"
10   }
11 }
```

Caller ID in this request can be used in order to check if there is an existing record in the CRM. Also, the caller dial input(DTMF) can be used as well.

# Client Response

## Response Parameters

Once it you have received the request from Voicenter and processed your automated process, a response needs to be formulated and sent back to Voicenter as part of the HTTP protocol process.

The response must contain the following parameters:

| Field Name  | Description   | Type    | Remarks   |
|-------------|---|---------|-----------|
| STATUS      | 0 - OK.<br>1 - Error has occurred on client's side.   | Integer | Mandatory |
| ACTION      | 1. GO_TO_LAYER - Will pass the call to a layer within a Voicenter IVR and continue as set in layer.<br><br>2. SAY – Will announce the caller with a data that was sent in this method and will then pass the call to a layer within a Voicenter IVR.<br>More info about the SAY method below.<br><br>3. DIAL - Will dial an external number or a Voicenter extension.<br>More info about the DIAL method below. | String  | Mandatory |
| Layer       | Layer number that the call will be redirected to.   | Integer | Mandatory |
| CALLER_NAME | Caller name that will be shown on the destination phone (Supported only for Voicenter extensions).  | String  | Optional  |
| CUSTOM_DATA | Internal CRM information about the client that can be passed for pop-up screen or CDR logs purposes.  | String  | Optional  |

## Client Response (Without Custom Data)

Below is an example of a response with GO\_TO\_LAYER action, the response is without custom data.

```
1  {
2    "STATUS":0,
3    "ACTION":"GO_TO_LAYER",
4    "LAYER":12
5  }
```

## Response With Custom Data

If you wish to send a response along with custom data can do so the following way:

```
1  {
2    "STATUS":0,
```

```
3      "ACTION": "GO_TO_LAYER",
4      "LAYER": 22,
5      "CALLER_NAME": "Shlomi",
6      "CUSTOM_DATA": {
7          "CRM_cleint_ID": "12345",
8          "Last_ticket_ID": "222",
9          "Last_representative": "John Doe"
10     }
11 }
```

## Case Example

You identify a caller by pressing their personal ID in a given Layer in the IVR (in this request example its "5").

Play the Layer recording (configured in the Cpanel incoming interface), asking the caller to press his ID (in the following request example "12345678").

After the caller has finished pressing, we will send a JSON request as followed:

## Case Example JSON Request

```
1  {
2    "METHOD": "IVR_LAYER_INPUT",
3    "DATA": {
4        "DID": "0722776772",
5        "CALLER_ID": "0501234567",
6        "IVR_UNIQUE_ID": "1bcd7954224861f85a2d70612f2",
7        "DTMF": "12345678",
8        "LAYER_ID": "5",
9        "PREVIOUS_LAYER_ID": "5"
10   }
11 }
```

The data on the "DTMF" field is the data that the caller pressed.

Next, we should get a Response indicating how the caller should be routed on the IVR.

## Case Example JSON Response

```
1  {
2    "STATUS": 0,
3    "ACTION": "GO_TO_LAYER",
4    "LAYER": 12,
5    "CALLER_NAME": null,
6    "CUSTOM_DATA": null
7  }
```

In this response, the caller was transferred to layer "12" in the IVR.

When we setup the integration, we create in advance, all the IVR scenarios (layers) according to the account preferences and business needs.

For example, layer "12" can be a layer indicating a successful caller identification and the caller will be directed to your representatives.

If the identification of the caller was not successful, the caller can be

redirected to layer "13" where we can ask to press the ID again, or any other option desired.

# Say Method

The SAY functionality of Voicenter external IVR service can be used to announce callers with dynamic data from your external system.

For example, you can inform your clients regarding the status of their order or let them know the amount of credits that currently left in their account, automatically through the IVR.

The initial request that is sent from Voicenter to your system is the same, as described above.

In order to use the "SAY" functionality we need to receive a response in the following format: POST- JSON.

Below is a table with the parameters you will need to use with a response example to provided afterwards.

## Response Parameters

| Field Name | Description  | Type    | Remarks   |
|------------|--|---------|-----------|
| STATUS     | 0 - OK.<br>1 - Error has occurred on client's side.  | String  | Mandatory |
| ACTION     | SAY_DIGITS   | String  | Mandatory |
| NEXT_LAYER | Layer number that the call will be redirected to after completing the announcement.  | Integer | Mandatory |
| LANGUAGE   | The language in which the client will hear the response:<br>1. HE – Hebrew.<br>2. EN – English.<br>3. AR – Arabic.<br>4. RU – Russian.<br>More languages can be supported upon request.  | String  | Mandatory |
| DATA       | A data array contains the data that needs to be played to the caller.<br>Types of data:<br>1. Digits (integer) - The IVR will announce the data digit-by-digit. Example: a phone number.<br>2. Number (integer) - The IVR will announce a number. Example: amount of credits left in the caller account.<br>3. Date (String) - The IVR will announce the date. Date in ISO 8601 format: YYYY-MM-DD. Example: 2019-12-14.<br>4. DateTime (String) - Announcement of date and time, ISO 8601 format, Example: 2019-12-14T10:12:59. | String  | Mandatory |

## Say Method Response Example

```
1  {
2    "STATUS":0,
3    "ACTION":"SAY_DIGITS",
4    "NEXT_LAYER":2,
5    "NEXT_LAYER_IF_FAILED":1,
6    "LANGUAGE":"EN",
7    "DATA":[
8      {
9        "RecordType":"Recording",
10       "Content":"9899115_20849570029.mp3"
11     }
12     {
13       "RecordType":"Digits",
14       "Content":"0501234567"
15     }
16   ]
17 }
```

```

16      {
17          "RecordType": "Number",
18          "Content": "112"
19      }
20      {
21          "RecordType": "Date",
22          "Content": "2019-12-14"
23      }
24      {
25          "RecordType": "DateTime",
26          "Content": "2019-12-14T10:12:59"
27      }
28      {
29          "RecordType": "Recording",
30          "Content": "98972915_209570029.mp3"
31      }
32  ]
33  }

```

## Say Method Response Explanation

The Digits, Number, Date and DateTime content will play in the language that was requested in the “LANGUAGE” field.  
 In this case, the data will be announced in English(“EN”).

The IVR will play the caller the data content in the order it was sent:

1. The caller will hear the recording - 9899115\_20849570029.mp3.
2. Then the IVR will play the digits “zero, five, zero, one, two, three, four, five, six, seven”.
3. After that the number "one hundred and twelve".
4. Date – “fourteenth of December two thousand nineteen".
5. Date and time – “fourteenth of December two thousand nineteen at ten and twelve minutes and fourteen seconds”.
6. The caller will hear the recording - 98972915\_20897670025.mp3.

After playing the data that was sent, the caller will be directed to the layer that was specified in NEXT\_LAYER. In this case, the call will be directed to layer 2.

## Dial Method

The DIAL functionality of Voicenter external IVR service can be used to direct calls to an external phone number or to a Voicenter extension.  
 For example, if your representatives are working with their personal mobile phones during night shifts and you wish to direct some calls to a specific representative mobile.  
 With the DIAL functionality you can easily do that.  
 The initial request that is sent from Voicenter to your system is the same, as described above.  
 In order to use the "SAY" functionality we need to receive a response in the following format: POST- JSON.  
 Below is a table with the parameters you will need to use with a response example to provided afterwards.

### Response Parameters

| Field Name | Description   | Type    | Remarks |
|------------|---|---------|---------|
| STATUS     | 0 - OK.<br>1 - Error has occurred on client's side. | Integer | Remarks |

| Field Name        | Description  | Type    | Remarks   |
|-------------------|--|---------|-----------|
| ACTION            | DIAL   | String  | Mandatory |
| CALLER_ID         | The phone number that will be shown to dialed destination. Example "0722776772".   | String  | Mandatory |
| CALLER_NAME       | A name that will be shown to the dialed destination.<br>Only if the destination is a Voicenter extension.  | String  | Optional  |
| MAX_CALL_DURATION | Total duration of the call(seconds). Actual call not include dialing duration.   | Integer | Mandatory |
| MAX_DIAL_DURATION | Total dialing duration of the call(seconds). Not include the actual answered call duration.  | Integer | Mandatory |
| NEXT_VO_ID        | In case of the destination is not reachable, the Voicenter IVR layer that the call will be directed to.<br>Important- it does not include busy\voicemail\dialed to the destination and the destination did not answer.<br>Only in cases of some kind of malfunction with the destination.      | Integer | Mandatory |
| RECORDING         | Whether to record the call? Accepts: "yes" or "no".<br>The default behavior is not to record the call.   | String  | Mandatory |
| TARGETS           | A data array that contains the requested call destination:<br>TYPE - accepts "PHONE" or "EXTENSION" (Voicenter Extensions). Value is case sensitive.<br>TARGET – accepts phone number or Voicenter extension SIP code.<br>Phone numbers outside Israel must be sent with international prefix. | String  | Mandatory |
| CUSTOM_DATA       | Internal CRM information about the client that can be passed for pop-up screen or CDR logs purposes.   | String  | Mandatory |

## Dial Method Response Example

```
1  {
2    "STATUS":0,
3    "ACTION":"DIAL",
4    "CALLER_ID":"0722776772",
5    "CALLER_NAME":"Voicenter",
6    "MAX_CALL_DURATION":1800,
7    "MAX_DIAL_DURATION":60,
8    "NEXT_VO_ID":15,
9    "RECORDING":"yes",
10   "TARGETS":[
11     {
12       "TYPE":"PHONE",
13       "TARGET":"0501234567"
14     }
15   ]
16   "CUSTOM_DATA":{
17     "CRM_cleint_ID":"12345",
18     "Last_ticket_ID":"222",
19     "Last_representative":"John Doe"
20   }
21 }
```



# External Layer API – Setting Up Layers

## General Description

Each layer in Voicenter IVR can direct the call to one of the following methods:

Call – dial 1 or more extensions or external destinations. Once the first destination answers – it stops ringing the other destinations.

Layer – transfer the caller to another IVR Layer.

Queue – Transfer the caller to a Queue in the system.

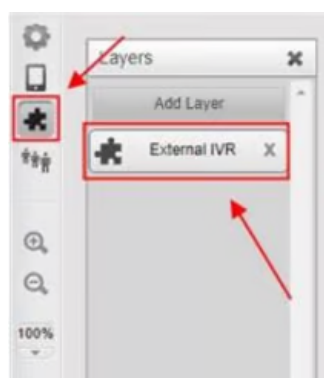
IVR structure and Layers settings are configurable from Voicenter Cpanel, in the “Incoming interface”.

Creating Layers to be used in the External IVR API is done in Voicenter Cpanel.

Once the IVR flow was determined and you finished creating the endpoint, please log in to Voicenter Cpanel.

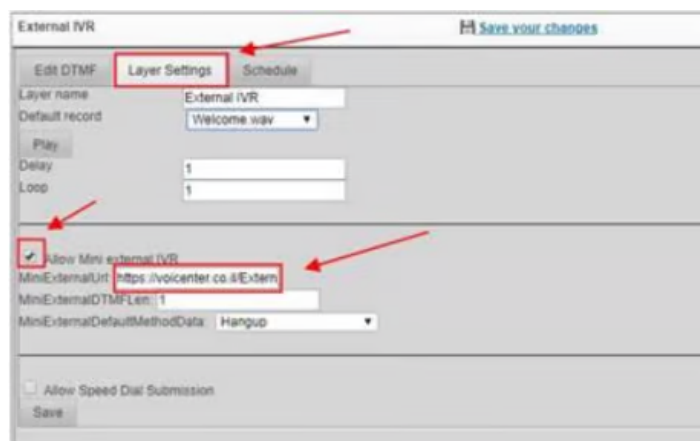
Go the Settings -> DIDs -> click the IVR column of the desired DID to open the Incoming interface.

In this page – press the Layers menu (the puzzle icon), and click the layer to configure:



Under the "Layer settings" tab check "Allow mini external IVR" checkbox and configure the following options:

**MiniExternalUrl** - Your system endpoint URL for this layer. This is where we are sending you the request:



**MiniExternalDTMFLen** - If your IVR requires an input from the caller(DTMF), determine what's the maximum digits allowed. Once this limit has been reached the Request will be sent.

**Delay** - When requesting a caller to dial digits – it usually takes a few seconds. Use this field to determine how many seconds after the announcement the system will wait before timing out(valid delay values are 1-9).

External IVR Save your changes

Edit DTMF   Layer Settings   Schedule

Layer name: External IVR  
 Default record: Welcome.wav  
 Play  
 Delay: 9  
 Loop: 1

---

☒ Allow Mini external IVR  
 MiniExternalUrl: https://voicenter.co.il/extern  
 MiniExternalDTMFLen: 10  
 MiniExternalDefaultMethodData: Default layer [IVR]

---

☐ Allow Speed Dial Submission  
 Save

The IVR will finish waiting for one of the following:

1. The caller response until it reaches the max digits input ("**MiniExternalDTMFLen**" field).
2. The delay timeout (seconds configured in "**Delay**" field) has passed.

**MiniExternalDefaultMethodData** - Select a failover layer in case the endpoint configured in MiniExternalUrl fails or times out.

If there is a problem to reach your endpoint, in order to prevent losing calls to simply hung-up because of malfunction, we usually configure a default layer so calls will be directed to it.

Layers

- External IVR X
- Default layer X

Edit DTMF   Layer Settings   Schedule

Layer name: External IVR  
 Default record: Welcome.wav  
 Play  
 Delay: 1  
 Loop: 1

---

☒ Allow Mini external IVR  
 MiniExternalUrl: https://voicenter.co.il/Extern  
 MiniExternalDTMFLen: 10  
 MiniExternalDefaultMethodData: Default layer [IVR]

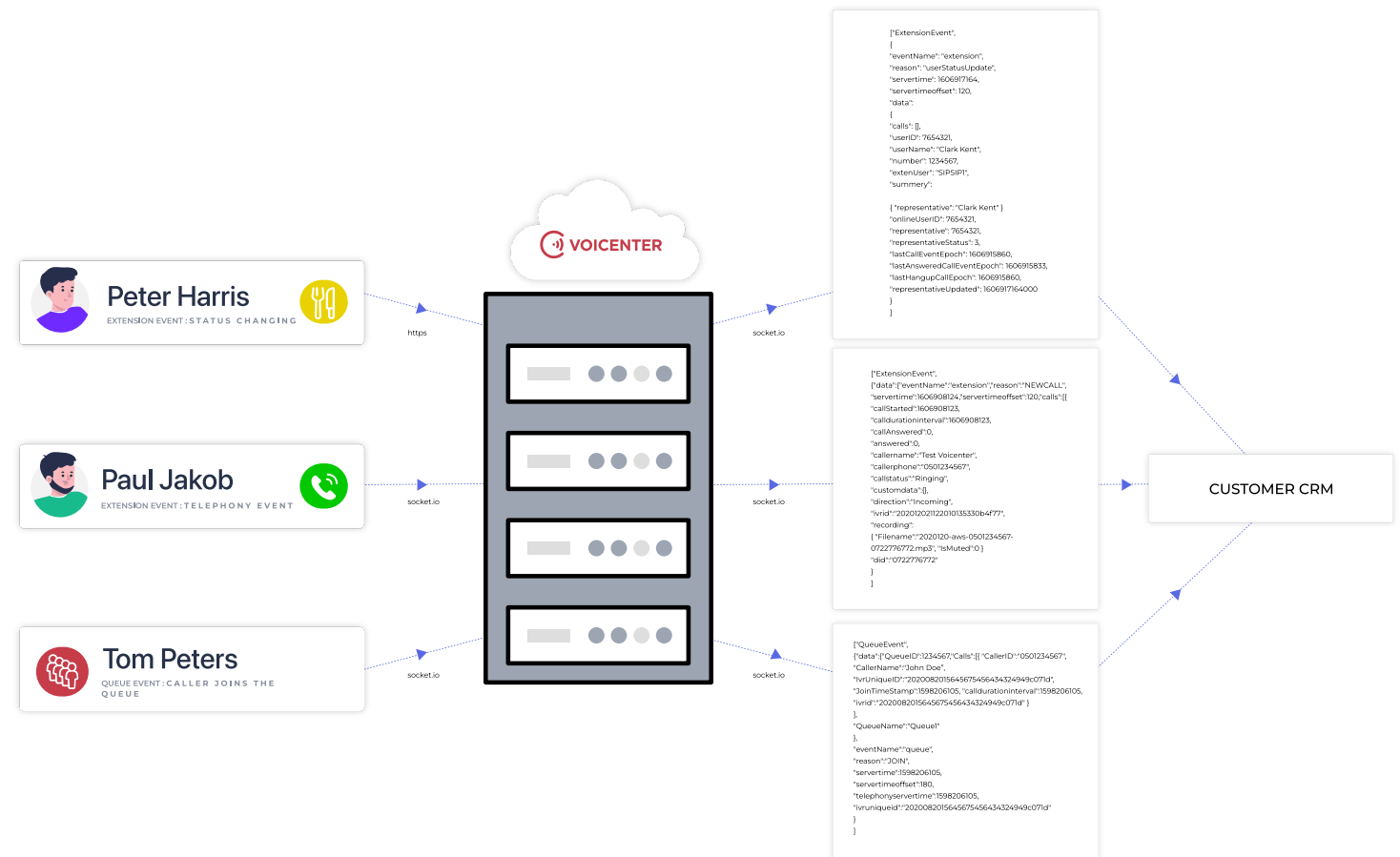
---

☐ Allow Speed Dial Submission  
 Save

# Real Time API

You can use the Real Time API to get events of calls and call status changes as well as user status changes in your organization.

By implementing The Real Time API in your system you can create pop-up screen, real-time dashboard and many more functionalities where you will be able to utilize your business logic along with you telephony activity.



## General Explanation

Voicenter RealTime API uses events (in JSON format) to showcase the telephony activity of your account.

The events are sent via socket communication with Voicenter APIs and backends.

You can use these event to create a popup or in case you are not using chrome browser the Real Time API will allow you to receive calling client's phone number in incoming calls.

This API suit for single page applications and using this API will allow you to do many other automated processes within your organization.

The Events SDK is used to communicate with Voicenter servers in order to receive real-time data via sockets.

Underneath, the events SDK uses socket.io to send and receive events.

You can access our Voicenter Real time Events SDK library by pressing [here](#) or NuGet library for VoicenterEventsSDK.NET [here](#).

## Login type

To use the Realtime API, your organization will need to login to the service.

There are two types of connections to the service - User/Account.

## User loginType

User - If you wish to receive events per user, you can use the credentials of each users that can be obtained via Voicenter Cpanel interface.

## User login Type example

```
1 let sdk = new EventsSDK({
2   loginType: 'user',
3   email: 'email@email.com',
4   password: 'password',
5 });
```

## Account Login Type

If you wish to receive events of your entire account, you can use your account credentials or account token (provided by Voicenter).

## Account login Type example

```
1 let sdk = new EventsSDK({
2   loginType: 'account',
3   username: 'username',
4   password: 'password',
5 });
```

## Account token loginType example

```
1 let sdk = new EventsSDK({
2   loginType: 'token',
3   useLoginApi: true,
4   token: 'token',
5 });
```

## HTML (WebSocket) connection example – Client Side

```
1 <html>
2 <head></head>
3 <body>
4   Status: <span id="status-txt">Disconnected</span>
5   <br/>
6   <br/>
7   <div id="response-body"></div>
8 </body>
9 </html>
10 <!-- Events SDK -->
11
12 <script src="https://cdn.voicenter.co/cdn/events-sdk/voicenter-events-sdk.umd.js"></script>
13
```

```

14 <script>
15 let sdk = new EventsSDK({
16 loginType: 'user',
17 email: 'email@email.com',
18 password: 'password',
19 });
20 sdk.init().then(function () {
21 // Success login event
22 sdk.on('loginSuccess', function (response) {
23 document.getElementById("status-txt").innerHTML = "Connected";
24 })
25 // Extension event
26 sdk.on('ExtensionEvent', function (response) {
27 console.log(response);
28 const cur = document.getElementById("response-body");
29 cur.innerHTML = cur.innerHTML + response.data.reason + "<br />";
30 })
31 // After logging in, receive all extensions statuses
32 sdk.on('AllExtensionsStatus', function (response) {
33 console.log(response);
34 })
35 });
36 </script>

```

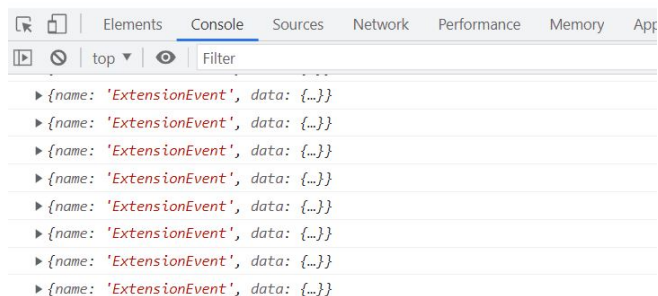
HTML sample project output example.

First glance of the initial page will reveal connection status and ongoing events of the user/account.

In order to view details of events, please go to your browser's diagnostic tools. (Normally done by pressing the F12 key).

Status: Connected

userStatusUpdate  
 userStatusUpdate  
 userStatusUpdate  
 NEWCALL  
 ANSWER  
 HANGUP  
 NEWCALL  
 NEWCALL  
 HANGUP  
 ANSWER  
 userStatusUpdate  
 NEWCALL  
 HOLD  
 ANSWER  
 HANGUP  
 HANGUP



## C# (WebSocket) connection example – Server Side

You can learn more from our demo [GitHub](#) repository.

```

1 using Newtonsoft.Json.Linq;
2 using System;
3 using System.Collections.Generic;
4 using System.Linq;

```

```

5    using System.Text;
6    using System.Threading.Tasks;
7    using VoicenterRealtimeAPI;
8    namespace ConsoleApp1
9    {
10   class Program
11   {
12   static void Main(string[] args)
13   {
14   //Logger
15   VoicenterRealtime voicenterRealtime = new VoicenterRealtime();
16   Logger.onLog += (object sender, VoicenterRealtimeLogger e) =>
17   {
18   Console.WriteLine(e.message);
19   };
20   //Listener
21   VoicenterRealtimeListener socket = voicenterRealtime.Token("XXXXX").Init();
22   socket.OnEvent += OnEventHandler;
23   socket.Listen();
24   Console.ReadLine();
25   }
26   }
27   }

```

This will the response after the connection:



## RealTime API – Events Types

The Voicenter web socket server has the following types of events sent to your client in real time.

While connected with the appropriate web socket containing the right permissions.

The data is sent in POST JSON format.

- Initial message
- loginSuccess
- loginStatus
- AllExtensionStatus
- QueueEvents
- ExtensionEvent

## Initial message

This type of event will be sent in the initial login connection request.

The sent data in this event describes the initial socket settings.

Mainly for Voicenter use.

| Field Name | Description | Type |
|------------|-------------|------|
|            |             |      |

| Field Name   | Description   | Type    |
|--------------|---|---------|
| sid          | Socket id - a unique code referring to this specific connection of socket that was established. | String  |
| Upgrades     | The list of possible transport upgrades.  | Array   |
| pingInterval | The Interval the of socket.io protocol.   | Integer |
| pingTimeout  | The timeout Interval of the socket.io protocol.   | Integer |

## Initial message Response

```
1 {
2   sid: "WF-eNmG0VbGkSn1wbD3n",
3   upgrades: [],
4   pingInterval: 60000,
5   pingTimeout: 180000
6 }
```

# loginSuccess

This type of event will be sent in the initial login connection request in case of a successful connection only. In case of wrong user name or password or token, you will receive a 401 (“Unauthorized”) or 500 (“Unexpected token”) http error.

## loginSuccess parameters

| Field Name       | Description                                      | Type    |
|------------------|--|---------|
| errorCode        | 0 - Connection was establishing successfully.    | Integer |
| errorDesc        | “Ok” - Connection was establishing successfully. | String  |
| servertime       | The Interval the of socket.io protocol.          | Integer |
| servertimeoffset | The timeout Interval of the socket.io protocol.  | Integer |

## loginSuccess parameters Example

```
1 {
2   errorCode: 0,
3   errorDesc: "OK"
4   servertime: 1597930812,
5   servertimeoffset: 180,
6 }
```

# loginStatus

This type of event will be sent only in case of a successful connection.  
The event describes the list of monitored queues on this active connection.  
The list of monitored queues will only be sent if the login connection was made with account credentials or account token.  
If the login connection type is by user credentials, only the user's extension assigned queues will return.

## loginStatus Parameters

| Field Name       | Description  | Type    |
|------------------|--|---------|
| errorCode        | 0 – ok. no errors.   | Integer |
| errorDesc        | “Ok” - Connection was establishing successfully.   | String  |
| servertime       | The Interval the of socket.io protocol.  | Integer |
| servertimeoffset | The timeout Interval of the socket.io protocol.  | Integer |
| Queues           | A list of the monitored queues.  | Array   |
| QueueID          | Voicenter queue ID.  | Integer |
| QueueName        | Voicenter queue name.  | String  |
| Calls            | A list of the current waiting calls in a specific queue.<br>If there are no waiting calls in the specific queue the array will be empty. | Array   |
| CallerID         | The caller caller ID.  | String  |
| CallerName       | The caller caller name.<br>In Voicenter calls you can show a string value for a call.<br>Not supported by all providers                  | String  |
| IvrUniqueID      | The call unique Voicenter ID.  | String  |
| Ivrid            | The call unique Voicenter ID (same as the field described above).  | String  |
| JoinTimeStamp    | When the call was entered to the queue (epoch).  | Integer |

## loginStatus Example

```
1  ["loginStatus",
2  {
3    "servertime":1597941727,
4    "queues":[
5      {
6        "Calls":[
7          {
8            "CallerID":"0501234567",
9            "CallerName":"John Doe",
10           "IvrUniqueID":"2020082015645675456434324949c071d",
11           "JoinTimeStamp":1597941709,
```



```
12
13   "ivrid": "2020082015645675456434324949c071d"
14   },
15 ],
16   "QueueID": 12345678,
17   "QueueName": "Queue1"
18 },
19 {
20   "Calls": [],
21   "QueueID": 12345679,
22   "QueueName": "Queue2"
23 }
24 ],
25   "errorCode": 0
26 }
27 ]
```

# AllExtensionStatus

This type of event will be sent in case of a successful connection only.

The event describes the list of active monitored extensions on this active connection.

The list of monitored extensions will only be sent if the login connection was made with account credentials or account token.

If the login connection type is by user credentials, the extensions list returns the current logged in user’s extension.

## AllExtensionStatus Table

| Field name       | Description   | Type    |
|------------------|---|---------|
| errorCode        | 0 – ok. no errors.  | Integer |
| errorDesc        | “Ok” - Connection was establishing successfully.  | String  |
| servertime       | The Interval the of socket.io protocol.   | Integer |
| servertimeoffset | The timeout Interval of the socket.io protocol.   | Integer |
| extensions       | A list of the monitored extensions.   | Array   |
| userID           | Voicenter user ID.<br>The user that the extensions is assigned to.<br>Each extension is assigned to a user.   | Integer |
| onlineUserID     | The user ID of the user that is currently logged on to the extension.<br>In case the representatives in your organization login in each shift to a different extension, this field will show the name of the representative that is currently working at the extension. | Integer |
| userName         | Voicenter user name.<br>The user that the extensions is assigned to.  | String  |
| number           | Voicenter extension ID.   | Integer |
| extenUser        | Sip username of the extension.  | String  |

| Field name                 | Description  | Type    |
|----------------------------|--|---------|
| representative             | The user ID of the user that is currently logged on to the extension.<br>In case the representatives in the organization login in each shift to a different extension, this field will show the name of the representative that is currently working at the extension.   | Integer |
| summery                    | In this array, we sent the name of the representative that is working with the extension.<br>In case no representative made a login action to the extension, the name of the user that the extension is assigned to will be sent.<br>Only the field which is sent in this array is “representative” with a String value. | Array   |
| representativeStatus       | The current login status ID of the connected user.<br>A value between 1-13 indicating the online user status.  | Integer |
| representativeUpdated      | The time of the last user status change.   | Integer |
| lastCallEventEpoch         | When was the last time a call was made in the extension, Epoch time.   | Integer |
| lastAnsweredCallEventEpoch | When was the last time a call was answered in the extension, Epoch time.   | Integer |
| lastHangupCallEpoch        | When was the last time a call was ended in the extension, Epoch time.  | Integer |
| Calls                      | A list of the current active calls in the extension.<br>If there are no dialing\ringing calls in the specific extension the list will be empty.  | Array   |
| CallerID                   | The caller caller ID.  | String  |
| CallerName                 | The caller caller name.<br>In Voicenter calls you can show a string value for a call.<br>Not supported by all providers.   | String  |
| IvrUniqueID                | The call unique Voicenter ID.  | String  |
| Ivrid                      | The call unique Voicenter ID (same as the field described above).  | String  |
| JoinTimeStamp              | When the call was entered to the queue (epoch).  | Integer |

## AllExtensionStatus Example

```
1  ["AllExtensionsStatus",
2  {
3  "servertime":1597945606,
4  "extensions":
5  [
6  {
7  "calls":[],
8  "userID":12345678,
9  "userName":"Clark Kent",
10 "number":9876543,
11 "onlineUserID":12345678,
12 "extenUser":"SIPSIP1",
```

```
13  "summery":
14  {
15    "representative":"Clark Kent"
16  },
17  "representative":0,
18  "representativeStatus":2,
19  "lastCallEventEpoch":1597933035,
20  "lastAnsweredCallEventEpoch":1545243466,
21  "representativeUpdated":0,
22  "peerStatus":"OK"
23  },
24  {
25    "calls":[],
26    "userID":12345679,
27    "userName":"Lois Lane",
28    "number":9876542,
29    "extenUser":"SIPSIP2",
30    "summery":
31    {
32      "representative":"Lois Lane"
33    },
34    "representative":12345679,
35    "representativeStatus":1,
36    "lastCallEventEpoch":1597944981,
37    "lastAnsweredCallEventEpoch":1597944981,
38    "representativeUpdated":1597944933000,
39    "peerStatus":"OK"
40  }
41  ],
42  "errorCode":0
43  }
44  ]
```

# QueueEvent

This type of event describes call entering and exiting a queue.  
It will only be sent if the initial socket login connection was made with account credentials or account token.  
If the login connection type is by user credentials, this type of event will not be sent.

## QueueEvent Parameters

| Field name       | Description  | Type    |
|------------------|--|---------|
| eventName        | "queue". Indication of the event type.   | String  |
| reason           | The reason this event was sent. For queue event there can be these reasons:<br>“JOIN” – A new call was entered to the queue and it is now waiting to be answered.<br>“EXIT” – When a waiting call is answered in one of the extensions.<br>“ABANDONED” - When a waiting call is not answered.<br>Either the caller chose to end the call or the call waiting time in the queue had reached the maximum waiting time. | String  |
| servertime       | The Interval the of socket.io protocol.  | Integer |
| servertimeoffset | The timeout Interval of the socket.io protocol.  | Integer |
| data             | A list of the queue event details.   | Array   |

| Field name    | Description   | Type    |
|---------------|---|---------|
| QueueID       | Voicenter queue ID.   | Integer |
| QueueName     | Voicenter queue name.   | String  |
| Calls         | A list of the current waiting calls in a specific queue.<br>If there are no waiting calls in the specific queue the list will be empty. | Array   |
| CallerID      | The caller caller ID.   | String  |
| CallerName    | The caller caller name.<br>In Voicenter calls you can show a string value for a call.<br>Not supported by all providers.                | String  |
| IvrUniqueID   | The call unique Voicenter ID.   | String  |
| Ivrid         | The call unique Voicenter ID (same as the field described above).   | String  |
| JoinTimeStamp | When the call was entered to the queue (epoch).   | Integer |

## QueueEvent JOIN Example

A call is entering a queue

```
1  ["QueueEvent",
2  {
3  "data":
4  {
5  "QueueID":1234567,
6  "Calls":
7  [
8  {
9  "CallerID":"0501234567",
10 "CallerName":"John Doe",
11 "IvrUniqueID":"2020082015645675456434324949c071d",
12 "JoinTimeStamp":1598206105,
13 "calldurationinterval":1598206105,
14 "ivrid":"2020082015645675456434324949c071d"
15 }
16 ],
17 "QueueName":"Queue1"
18 },
19 "eventName":"queue",
20 "reason":"JOIN",
21 "servertime":1598206105,
22 "servertimeoffset":180,
23 "telephonyservertime":1598206105,
24 "ivruniqueid":"2020082015645675456434324949c071d"
25 }
26 ]
```

## QueueEvent EXIT Example

A call is answered by an extension (a call is exiting the queue)

```
1  ["QueueEvent",
2  {
3  "data":
4  {
5  "QueueID":1234567,
6  "Calls":[],
7  "QueueName":"Queue1"
8  },
9  "eventName":"queue",
10 "reason":"EXIT",
11 "servertime":1598206119,
12 "servertimeoffset":180,
13 "telephonyservertime":1598206119,
14 "ivruniqueid":"2020082015645675456434324949c071d"
15 }
16 ]
```

## QueueEvent ABANDONED Example

The caller chose to end the call before his call was answered

```
1  ["QueueEvent",
2  {
3  "data":
4  {
5  "QueueID":1234567,
6  "Calls":[],
7  "QueueName":"Queue1"
8  },
9  "eventName":"queue",
10 "reason":"ABANDONED",
11 "servertime":1598206251,
12 "servertimeoffset":180,
13 "telephonyservertime":1598206251,
14 "ivruniqueid":"2020082015645675456434324949c0712"
15 }
16 ]
```

## ExtensionEvent

This type of event describes call status(dialing\talking\hung-up) in the account extensions.  
If the initial socket login connection was made with user credentials, only the user assigning extension\s will receive events.  
If the initial socket login connection was made with account credentials or account token, all the extensions events will be sent.

### ExtensionEvent Parameters

| Field name | Description                                   | Type   |
|------------|---|--------|
| eventName  | "extension".<br>Indication of the event type. | String |

| Field name       | Description  | Type    |
|------------------|--|---------|
| reason           | <p>The reason this event was sent.</p> <p>“NEWCALL” – A new call is ringing or dialing from an extension.</p> <p>“ANSWER” – A call was answered at an extension.</p> <p>“HOLD” – A call was placed on hold.</p> <p>“UNHOLD” – A call is no longer on hold.</p> <p>“HANGUP” – When a call was ended. It is not necessarily means that the call was answered.</p> <p>“userStatusUpdate” – When the extension online user updated his\her user status(Login, Break, Logout, etc.)</p>   | String  |
| cause            | <p>This field will only be sent if the event “reason”= “HANGUP” and will indicate the cause of the hangup</p> <p>“Normal hangup” – If a call rang at an extension and the call was ended. It is not necessarily means that the call was answered.</p> <p>“Answered elsewhere” – If the call rang simultaneously at several extensions and was answered, an “HANGUP” event will send to the all other extensions who did not answered the call.</p> <p>“Call Rejected” – In case a leg1 click2call call was not answered.</p> | String  |
| servertime       | The Interval the of socket.io protocol.  | Integer |
| servertimeoffset | The timeout Interval of the socket.io protocol.  | Integer |
| Data             | A list of the extension event details.   | Array   |
| Calls            | <p>A list of the current active calls in the extension.</p> <p>If there are no dialing\ringing calls in the specific extension the list will be empty.</p>   | Array   |
| currentCall      | <p>The “currentCall” data indicate which call the current event relates to. In case there is more than one call on the extension, this array will indicate for which specific call this event is sent.</p> <p>For example, if the extension is talking at a particular moment with a caller and during that time another waiting call is entered. For the new entered call, a new event will be send.</p>  | Array   |
| answered         | <p>Whether the call was answered.</p> <p>0 – Not answered.</p> <p>1 – Answered.</p>  | Boolean |
| callAnswered     | <p>Time in Epoch. When the call was answered.</p> <p>In call did not answered yet the sent value will be 0.</p>  | Integer |
| callStarted      | Time in Epoch. When the call was started.  | Integer |
| callername       | <p>What string value will appear at the extension.</p> <p>In Voicenter you can set a text value that will appear in when the call in ringing\dialing.</p> <p>This data will appear only in Voicenter extensions.</p>   | String  |
| callerphone      | The caller or callee phone number.   | String  |

| Field name   | Description  | Type    |
|--------------|--|---------|
| callstatus   | What is the current call status?<br>“Ringing”- In case of an incoming call,<br>“Dialing”- In case of an Outgoing call,<br>“Talking” – In case a call was answered at an extension,<br>"Spying" - In case an extension is listening to another extension in the organization. | String  |
| customdata   | In case custom variables were assigned to the call.  | Array   |
| did          | The phone number that the caller dialed to.<br>This field will only send value in incoming calls.  | String  |
| direction    | The direction of the call:<br>“Incoming”,<br>“Outgoing”,<br>“Spy” – In case, an extension is listening to another extension in the organization.<br>“Click2call”.  | String  |
| c2cdirection | This field will only be sent in case the call type consists of two parts (Leg1, Leg2).<br>Two values are sent:<br>1 - Leg1.<br>2 - Leg2.   | Integer |
| ivrid        | The call unique identifier.  | String  |
| recording    | The call recording data details.   | Array   |
| Filename     | The name of the call recording file.<br>Will be sent if the call was supposed to be recorded.<br>In case of an outgoing call, the data will be sent if the call was answered(it will not be sent in “NEWCALL” type event).   | String  |
| IsMuted      | Indicate whether the call recording was muted.<br>With the Voicenter Mute API and softphone, representatives can pause the recording of the call and unpause it.<br>0 – The call was not muted.<br>1 – The call was muted.   | String  |
| extenUser    | The current extension SIP code.  | String  |
| userID       | Voicenter user ID.<br>The user that the extensions is assigned to.<br>Each extension is assigned to a user.  | Integer |
| onlineUserID | The user ID of the user that is currently logged on to the extension.<br>In case the representatives in the organization login in each shift to a different extension, this field will show the name of the representative that is currently working at the extension.       | Integer |
| userName     | Voicenter user name.<br>The user that the extensions is assigned to.   | String  |
| number       | Voicenter extension ID.  | Integer |

| Field name                 | Description  | Type    |
|----------------------------|--|---------|
| extenUser                  | The SIP user name of the relevant extension.   | String  |
| representative             | The user ID of the user that is currently logged on to the extension.<br>In case the representatives in the organization login in each shift to a different extension, this field will show the name of the representative that is currently working at the extension. | Integer |
| representativeStatus       | The current login status ID of the connected user.<br>A value between 1-13 indicating the online user status.  | Integer |
| representativeUpdated      | The time of the last user status change.   | Integer |
| lastCallEventEpoch         | When was the last time a call was made in the extension in Epoch time.   | Integer |
| lastAnsweredCallEventEpoch | When was the last time a call was answered in the extension in Epoch time.   | Integer |
| lastHangupCallEpoch        | When was the last time a call was ended in the extension in Epoch time.  | Integer |
| callerID                   | For incoming calls, the caller caller ID.<br>For outgoing calls the destination caller ID.<br>This field will be send on “reason”= “HANGUP” calls.   | String  |
| ivruniqueid                | The call unique identifier.<br>This field will be send on “reason”：“HANGUP” calls.<br>Same as the field – “ivrid”.   | String  |
| queueID                    | Voicenter queue ID.<br>This field will be sent only if the call was directed through Voicenter queue service.  | Integer |
| campaignID                 | Voicenter campaign ID.<br>This field will be sent only if the call was made by Voicenter automatic dialer service.   | Integer |

# ExtensionEvent Examples

## ExtensionEvent - Incoming NEWCALL

An incoming call is ringing at an extension – reason: NEWCALL

```
1  ["ExtensionEvent",
2  {"data":
3  {
4  "eventName":"extension",
5  "reason":"NEWCALL",
6  "servertime":1606908124,
7  "servertimeoffset":120,
8  "calls":[
9  {
10 "callStarted":1606908123,
11 "calldurationinterval":1606908123,
12 "callAnswered":0,
```



```

13  "answered":0,
14  "callername":"Test Voicenter",
15  "callerphone":"0501234567",
16  "callstatus":"Ringing",
17  "customdata":{},
18  "direction":"Incoming",
19  "ivrid":"202012021122010135330b4f77",
20  "recording":
21  {
22  "Filename":"2020120-aws-0501234567-0722776772.mp3",
23  "IsMuted":0
24  },
25  "did":"0722776772"
26  }
27  ],
28  "userID":7654321,
29  "userName":"Clark Kent",
30  "number":1234567,
31  "extenUser":"SIPSIP1",
32  "summery":{
33  "representative":"Clark Kent"
34  },
35  "onlineUserID":7654321,
36  "representative":7654321,
37  "representativeStatus":1,
38  "lastCallEventEpoch":1606908124,
39  "lastAnsweredCallEventEpoch":1606847063,
40  "lastHangupCallEpoch":1606905028,
41  "representativeUpdated":1606763622000,
42  "currentCall":
43  {
44  "callStarted":1606908123,
45  "calldurationinterval":1606908123,
46  "callAnswered":0,
47  "answered":0,
48  "callername":"Test Voicenter",
49  "callerphone":"0501234567",
50  "callstatus":"Ringing",
51  "customdata":{},
52  "direction":"Incoming",
53  "ivrid":"202012021122010135330b4f77",
54  "recording":
55  {
56  "Filename":"2020120-aws-0501234567-0722776772.mp3",
57  "IsMuted":0
58  },
59  "did":"0722776772"
60  }
61
62  }
63  }
64  ]

```

## ExtensionEvent - Incoming ANSWER

An incoming call is answered at an extension – reason: ANSWER

```

1  ["ExtensionEvent",
2  {
3  "data": {
4  "eventName": "extension",

```

```

5      "reason": "ANSWER",
6      "servertime": 1606908124,
7      "servertimeoffset": 120,
8      "calls": [{
9          "callStarted": 1606908123,
10         "calldurationinterval": 1606908123,
11         "callAnswered": 1606908134,
12         "answered": 1,
13         "callername": "Test Voicenter",
14         "callerphone": "0501234567",
15         "callstatus": "Talking",
16         "customdata": {},
17         "direction": "Incoming",
18         "ivrid": "202012021122010135330b4f77",
19         "recording": {
20             "Filename": "2020120-aws-0501234567-0722776772.mp3",
21             "IsMuted": 0
22         },
23         "did": "0722776772"
24     }],
25     "userID": 7654321,
26     "userName": "Clark Kent",
27     "number": 1234567,
28     "extenUser": "SIPSIP1",
29     "summery": {
30         "representative": "Clark Kent"
31     },
32     "onlineUserID": 7654321,
33     "representative": 7654321,
34     "representativeStatus": 1,
35     "lastCallEventEpoch": 1606908124,
36     "lastAnsweredCallEventEpoch": 1606847063,
37     "lastHangupCallEpoch": 1606905028,
38     "representativeUpdated": 1606763622000,
39     "currentCall": {
40         "callStarted": 1606908123,
41         "calldurationinterval": 1606908123,
42         "callAnswered": 1606908134,
43         "answered": 1,
44         "callername": "Test Voicenter",
45         "callerphone": "0501234567",
46         "callstatus": "Talking",
47         "customdata": {},
48         "direction": "Incoming",
49         "ivrid": "202012021122010135330b4f77",
50         "recording": {
51             "Filename": "2020120-aws-0501234567-0722776772.mp3",
52             "IsMuted": 0
53         },
54         "did": "0722776772"
55     }
56 }
57 }
58 ]

```

## ExtensionEvent - Incoming HANGUP

An incoming call was ended after it was answered – reason: HANGUP

```

1      ["ExtensionEvent",
2      {"data":

```

```

3      {
4          "eventName":"extension",
5          "reason":"HANGUP",
6          "cause": "Normal hangup",
7          "callerID": "0501234567",
8          "ivruniqueid": "202012021122010135330b4f77",
9          "servertime":1606908139,
10         "servertimeoffset":120,
11         "calls":[],
12         "userID":7654321,
13         "userName":"Clark Kent",
14         "number":1234567,
15         "extenUser":"SIPSIP1",
16         "summery":{
17             "representative":"Clark Kent"
18         },
19         "onlineUserID":7654321,
20         "representative":7654321,
21         "representativeStatus":1,
22         "lastCallEventEpoch":1606908139,
23         "lastAnsweredCallEventEpoch":1606908139,
24         "lastHangupCallEpoch":1606908139,
25         "representativeUpdated":1606763622000,
26         "currentCall":
27         {
28             "callStarted":1606908123,
29             "calldurationinterval":1606908123,
30             "callAnswered":1606908134,
31             "answered":1,
32             "callername":"Test Voicenter",
33             "callerphone":"0501234567",
34             "callstatus":"Talking",
35             "customdata":{},
36             "direction":"Incoming",
37             "ivrid":"202012021122010135330b4f77",
38             "recording":
39             {
40                 "Filename":"2020120-aws-0501234567-0722776772.mp3",
41                 "IsMuted":0
42             },
43             "did":"0722776772"
44         }
45     }
46 }
47 }
48 ]

```

## ExtensionEvent - Click2call call Leg1 NEWCALL

An outgoing Click2call call. Leg1 – reason: NEWCALL

```

1      ["ExtensionEvent",
2      {"data":
3      {
4          "eventName":"extension",
5          "reason":"NEWCALL",
6          "servertime":1606913080,
7          "servertimeoffset":120,
8          "calls":[
9          {

```

```

10  "callStarted":1606913080,
11  "calldurationinterval":1606913080,
12  "callAnswered":0,
13  "answered":0,
14  "callername":"0501234567",
15  "callerphone":"0501234567",
16  "callstatus":"Dialing",
17  "customdata":{},
18  "direction":"Click2Call",
19  "ivrid":"20201200013quvnhxku5v",
20  "recording":
21  {
22  "Filename":"2020120-aws-SIPSIP1-972501234567.mp3",
23  "IsMuted":0
24  },
25  "did":"",
26  "c2cdirection":"1"
27  }
28  ],
29  "userID":7654321,
30  "userName":"Clark Kent",
31  "number":1234567,
32  "extenUser":"SIPSIP1",
33  "summery":{
34  "representative":"Clark Kent"
35  },
36  "onlineUserID":7654321,
37  "representative":7654321,
38  "representativeStatus":1,
39  "lastCallEventEpoch":1606913081,
40  "lastAnsweredCallEventEpoch":1606908139,
41  "lastHangupCallEpoch":1606908139,
42  "representativeUpdated":1606763622000,
43  "currentCall":
44  {
45  "callStarted":1606913080,
46  "calldurationinterval":1606913080,
47  "callAnswered":0,
48  "answered":0,
49  "callername":"0501234567",
50  "callerphone":"0501234567",
51  "callstatus":"Dialing",
52  "customdata":{},
53  "direction":"Click2Call",
54  "ivrid":"20201200013quvnhxku5v",
55  "recording":
56  {
57  "Filename":"2020120-aws-0501234567-0722776772.mp3",
58  "IsMuted":0
59  },
60  "did":"",
61  "c2cdirection":"1"
62  }
63  }
64  }
65  }
66  ]

```

## ExtensionEvent - Click2call call Leg1 ANSWER

An outgoing Click2call call. Leg1 – reason: ANSWER

```
1  ["ExtensionEvent",
2  {"data":
3  {
4  {
5  "eventName":"extension",
6  "reason":"ANSWER",
7  "servertime":1606913082,
8  "servertimeoffset":120,
9  "calls":[
10 {
11 "callStarted":1606913080,
12 "calldurationinterval":1606913080,
13 "callAnswered":1606913082,
14 "answered":1,
15 "callername":"0501234567",
16 "callerphone":"0501234567",
17 "callstatus":"Talking",
18 "customdata":{},
19 "direction":"Click2Call",
20 "ivrid":"20201200013quvnhxku5v",
21 "recording":
22 {
23 "Filename":"2020120-aws-SIPSIP1-972501234567.mp3",
24 "IsMuted":0
25 },
26 "did":"",
27 "c2cdirection":"1"
28 }
29 ],
30 "userID":7654321,
31 "userName":"Clark Kent",
32 "number":1234567,
33 "extenUser":"SIPSIP1",
34 "summery":{
35 "representative":"Clark Kent"
36 },
37 "onlineUserID":7654321,
38 "representative":7654321,
39 "representativeStatus":1,
40 "lastCallEventEpoch":1606913081,
41 "lastAnsweredCallEventEpoch":1606908139,
42 "lastHangupCallEpoch":1606908139,
43 "representativeUpdated":1606763622000,
44 "currentCall":
45 {
46 "callStarted":1606913080,
47 "calldurationinterval":1606913080,
48 "callAnswered":1606913082,
49 "answered":1,
50 "callername":"0501234567",
51 "callerphone":"0501234567",
52 "callstatus":"Talking",
53 "customdata":{},
54 "direction":"Click2Call",
55 "ivrid":"20201200013quvnhxku5v",
56 "recording":
57 {
58 "Filename":"2020120-aws-0501234567-0722776772.mp3",
59 "IsMuted":0
60 },
61 "did":"",
62 "c2cdirection":"1"
63 }
64 }
65 }
66 }
```

## ExtensionEvent - Click2call call Leg2 NEWCALL

An outgoing Click2call call. Leg2 – reason: NEWCALL

```
1  ["ExtensionEvent",
2  {"data":
3  {
4  "eventName":"extension",
5  "reason":"NEWCALL",
6  "servertime":1606913083,
7  "servertimeoffset":120,
8  "calls":[
9  {
10 "callStarted":1606913080,
11 "calldurationinterval":1606913080,
12 "callAnswered":0,
13 "answered":0,
14 "callername":"0501234567",
15 "callerphone":"0501234567",
16 "callstatus":"Talking",
17 "customdata":{},
18 "direction":"Click2Call",
19 "ivrid":"20201200013quvnhxku5v",
20 "recording":
21 {
22 "Filename":"2020120-aws-SIPSIP1-972501234567.mp3",
23 "IsMuted":0
24 },
25 "did":"",
26 "c2cdirection":"2"
27 }
28 ],
29 "userID":7654321,
30 "userName":"Clark Kent",
31 "number":1234567,
32 "extenUser":"SIPSIP1",
33 "summery":{
34 "representative":"Clark Kent"
35 },
36 "onlineUserID":7654321,
37 "representative":7654321,
38 "representativeStatus":1,
39 "lastCallEventEpoch":1606913083,
40 "lastAnsweredCallEventEpoch":1606913082,
41 "lastHangupCallEpoch":1606908139,
42 "representativeUpdated":1606763622000,
43 "currentCall":
44 {
45 "callStarted":1606913080,
46 "calldurationinterval":1606913080,
47 "callAnswered":0,
48 "answered":0,
49 "callername":"0501234567",
50 "callerphone":"0501234567",
51 "callstatus":"Talking",
52 "customdata":{},
53 "direction":"Click2Call",
54 "ivrid":"20201200013quvnhxku5v",
55 "recording":
```

```

56 {
57   "Filename": "2020120-aws-0501234567-0722776772.mp3",
58   "IsMuted": 0
59 },
60 "did": "",
61 "c2cdirection": "2"
62 }
63
64 }
65 }
66 ]

```

## ExtensionEvent - Click2call call Leg2 ANSWER

An outgoing Click2call call. Leg2 – reason: ANSWER

```

1  ["ExtensionEvent",
2  {"data":
3  {
4    "eventName": "extension",
5    "reason": "ANSWER",
6    "servertime": 1606913090,
7    "servertimeoffset": 120,
8    "calls": [
9    {
10     "callStarted": 1606913080,
11     "calldurationinterval": 1606913080,
12     "callAnswered": 1606913090,
13     "answered": 1,
14     "callername": "0501234567",
15     "callerphone": "0501234567",
16     "callstatus": "Talking",
17     "customdata": {},
18     "direction": "Click2Call",
19     "ivrid": "20201200013quvnhxku5v",
20     "recording":
21     {
22       "Filename": "2020120-aws-SIPSIP1-972501234567.mp3",
23       "IsMuted": 0
24     },
25     "did": "",
26     "c2cdirection": "2"
27   }
28 ],
29   "userID": 7654321,
30   "userName": "Clark Kent",
31   "number": 1234567,
32   "extenUser": "SIPSIP1",
33   "summery": {
34     "representative": "Clark Kent"
35   },
36   "onlineUserID": 7654321,
37   "representative": 7654321,
38   "representativeStatus": 1,
39   "lastCallEventEpoch": 1606913083,
40   "lastAnsweredCallEventEpoch": 1606913082,
41   "lastHangupCallEpoch": 1606908139,
42   "representativeUpdated": 1606763622000,
43   "currentCall":
44   {
45     "callStarted": 1606913080,

```

```

46 "calldurationinterval":1606913080,
47 "callAnswered":1606913090,
48 "answered":1,
49 "callername":"0501234567",
50 "callerphone":"0501234567",
51 "callstatus":"Talking",
52 "customdata":{},
53 "direction":"Click2Call",
54 "ivrid":"20201200013quvnhxku5v",
55 "recording":
56 {
57 "Filename":"2020120-aws-0501234567-0722776772.mp3",
58 "IsMuted":0
59 },
60 "did":"",
61 "c2cdirection":"2"
62 }
63
64 }
65 }
66 ]

```

## ExtensionEvent - Click2call call Leg2 HANGUP

An outgoing Click2call call. Leg2 – reason: HANGUP

```

1  ["ExtensionEvent",
2  {"data":
3  {
4  "eventName":"extension",
5  "reason":"HANGUP",
6  "servertime":1606913094,
7  "servertimeoffset":120,
8  "callerID": "0501234567",
9  "ivruniqueid": "20201200013quvnhxku5v",
10 "cause": "Normal hangup",
11 "calls":[],
12 "userID":7654321,
13 "userName":"Clark Kent",
14 "number":1234567,
15 "extenUser":"SIPSIP1",
16 "summery":{
17 "representative":"Clark Kent"
18 },
19 "onlineUserID":7654321,
20 "representative":7654321,
21 "representativeStatus":1,
22 "lastCallEventEpoch":1606913083,
23 "lastAnsweredCallEventEpoch":1606913082,
24 "lastHangupCallEpoch":1606908139,
25 "representativeUpdated":1606763622000,
26 "currentCall":
27 {
28 "callStarted":1606913080,
29 "calldurationinterval":1606913080,
30 "callAnswered":1606913090,
31 "answered":1,
32 "callername":"0501234567",
33 "callerphone":"0501234567",
34 "callstatus":"Talking",
35 "customdata":{},
36 "direction":"Click2Call",

```



```

37   "ivrid": "20201200013quvnhxku5v",
38   "recording":
39   {
40     "Filename": "2020120-aws-0501234567-0722776772.mp3",
41     "IsMuted": 0
42   },
43   "did": "",
44   "c2cdirection": "2"
45 }
46
47 }
48 }
49 ]

```

## A user status change – reason: `userStatusUpdate`

The representative change his\her online status to status “3” (Lunch).

The list of all available user statuses can be found in the Voicenter “Login\Logout” API documentation.

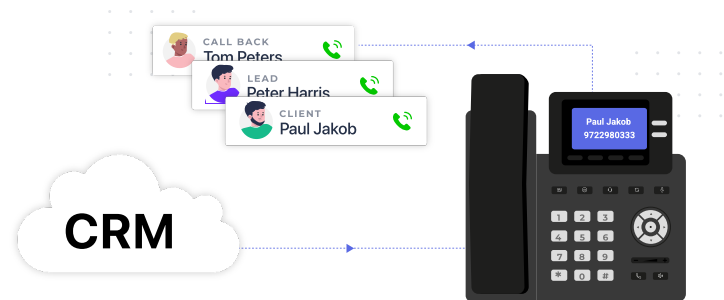
```

1   ["ExtensionEvent",
2   {
3     "eventName": "extension",
4     "reason": "userStatusUpdate",
5     "servertime": 1606917164,
6     "servertimeoffset": 120,
7     "data":
8     {
9       "calls": [],
10      "userID": 7654321,
11      "userName": "Clark Kent",
12      "number": 1234567,
13      "extenUser": "SIPSIP1",
14      "summery": {
15        "representative": "Clark Kent"
16      },
17      "onlineUserID": 7654321,
18      "representative": 7654321,
19      "representativeStatus": 3,
20      "lastCallEventEpoch": 1606915860,
21      "lastAnsweredCallEventEpoch": 1606915833,
22      "lastHangupCallEpoch": 1606915860,
23      "representativeUpdated": 1606917164000
24    }
25  }
26 ]

```

# Dialer API

Using this API You will be able to upload or remove call destinations, stop and start campaigns, getting your organization campaigns list, and many more features this API provides.



## General Explanation

The dialer api will allow you to connect to your customers faster than ever by allowing you to perform many dialer related actions straight through your CRM system.

This way you can manage the activity of your dialer campaigns via your CRM, in only one interface.

Some of the actions you can perform are:

- Get a list of the campaigns in your Voicenter account.

- Add or Remove destinations from a campaign.

- Add or remove agents to a campaign.

- Stop or start a campaign.

- Update your IVR Dialer campaign.

When using the dialer API, firstly you will need to use the GetCampaignList method where you will get the active campaigns in your Voicenter account.

## GetCampaignList

Returns all automatic dialer campaigns.

Voicenter auto dialer service allows users to manage several separate sessions called "Campaigns".

Each "Campaign" can have its own settings and can dial to different destinations simultaneously.

Most of this API methods requires a specific campaign code.

With GetCampaignList you can get all this data in one place.

## URI

<https://api.voicenter.com/ForwardDialer/Dialer/GetCampaignList>

## Acceptable Request Types

POST-JSON

GET

## Possible Response Formats

POST-JSON

## Client Request Parameters

| Field name | Description                                 | Type   | Remarks   |
|------------|---|--------|-----------|
| Code       | Personal identifier, provided by Voicenter. | String | Mandatory |

Client Request Example (GET)

```
1 | https://api.voicenter.com/ForwardDialer/Dialer/GetCampaignList?Code=XXXXXXXXXXXXX
```

Client Request Example (POST - JSON)

```
1 | {
2 |   "Code" : "XXXXXXXXXXXXX"
3 | }
```

Voicenter Response Parameters

| Field name        | Description   | Type    |
|-------------------|---|---------|
| ErrorCode         | Response status:<br>"0" - No Errors.  | Integer |
| Description       | Response status description:<br>"OK" - No errors.   | String  |
| Data              | JSON array type.<br>List of the requested automatic dialer campaigns, both active and non active campaigns. | Array   |
| Name              | The Voicenter campaign name.  | String  |
| StatusName        | Campaign status:<br>1. "Enabled" - Active Campaign.<br>2. "Disabled" - Stopped Campaign.                    | String  |
| TotalPendingCalls | The sum of waiting records destinations to be dial.   | Integer |
| MaxPriority       | Returns the highest priority value that was set for a call destination record in the campaign.              | Integer |
| MinPriority       | Returns the lowest priority value that was set for a call destination record in the campaign.               | Integer |

| Field name         | Description  | Type    |
|--------------------|--|---------|
| TotalAwaitingCalls | Relevant only for IVR dialer campaigns.<br>The sum of waiting records destinations to be dial at the IVR dialer service.<br>This field is more recommended to use then "TotalPendingCalls" for IVR dialer campaign.<br>In case of agent dialer campaign the returned value will be "null". | Integer |
| Code               | The campaign unique identifier code.<br>Needed for the auto dialer API methods.  | String  |

## Voicenter Response Example

```
1  {
2    "Data": [
3      {
4        "Name": "Campaign1",
5        "StatusName": "Enabled",
6        "TotalPendingCalls": 23,
7        "MaxPriority": 1,
8        "MinPriority": 1,
9        "TotalAwaitingCalls": 23,
10       "Code": "XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
11     }
12     {
13       "Name": "Campaign2",
14       "StatusName": "Enabled",
15       "TotalPendingCalls": 19,
16       "MaxPriority": 52,
17       "MinPriority": 1,
18       "TotalAwaitingCalls": null,
19       "Code": "XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
20     }
21   ]
22   "ErrorCode": 0,
23   "Description": "OK"
24 }
```

# AddCall

Using this method you will be able to add a single destination to your campaign.  
In case you wish to add multiple destinations to a campaign, please use the next method: AddCallsBulk.

## URI

https://api.voicenter.com/ForwardDialer/Dialer/AddCall

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

- 1.POST-JSON

## Client Request Parameters

| Field name      | Description  | Type    | Remarks   |
|-----------------|--|---------|-----------|
| Campaign        | Campaign unique identifier, provided by GetCampaignList (7.1) method.  | String  | Mandatory |
| Target          | Call target destination phone number.<br>Must be sent with the international prefix (for Israel destinations not mandatory).   | String  | Mandatory |
| CustomerName    | Call target destination name.  | String  | Optional  |
| CallerID        | The phone number that will be display to the destination.<br>Must be phone number in your Voicenter account.   | String  | Optional  |
| Priority        | Call target with higher priority values will dial before lower ones.   | Integer | Optional  |
| CustomData      | You may set your own fields and values that will be associated with each call target destination.<br>This data can be used for pop up and call log.  | Array   | Optional  |
| OriginateTime   | You can set calls to be dial at future date.<br>EPOCH time format.<br>In case this field is not sent in the request, the call will dial as soon as possible.<br>Must be send together with "IsDateLocal" field.  | Integer | Optional  |
| IsDateLocal     | Whether the future call date will be in local time as set in the Voicenter Cpanel.<br>In case "false" value was set, the call will be dialed at GMT 0.<br>It is recommended to send "true" value.<br>Must be send together with "OriginateTime" field. | Boolean | Optional  |
| IgnoreDncStatus | In case you activated Voicenter Do-Not-Call-Me service, a service that checks whether to allow calling Israeli destinations, by adding this field to your Auto-Dialer service requests, you will be allowed to make calls to restricted destinations.  | Boolean | Optional  |

## Client Request Example (GET)

```
1 https://api.voicenter.com/ForwardDialer/Dialer/AddCall?Campaign=XXXXXXXXXXXXXXXXXXXX&Target=0501234567&CustomerName=JohnDoe
```

## Client Request Example (POST - JSON)

```
1 {
2   "Campaign": "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
3   "Target": "0501234567",
4   "CustomerName": "John Doe",
```

```
5      "CallerID": "0722776772",
6      "Priority": 42,
7      "OriginateTime": 1602750000,
8      "IsDateLocal": "true",
9      "IgnoreDncStatus": "true",
10     "CustomData": {
11         "var_LeadID": 1234567,
12         "var_LeadCampaign": "Facebook"
13     }
14 }
```

## Voicenter Response Parameters

| Field name  | Description  | Type    |
|-------------|--|---------|
| ErrorCode   | Response status:<br>"0" – No errors.<br>"1" – Invalid campaign code.<br>"2" – Target missing.<br>"-2" – Number is invalid.   | Integer |
| Description | Response status description:<br>"OK" – No errors.<br>"Invalid campaign code" – The sent campaign identifier is invalid.<br>"Campaign code missing" - The "Campaign" field is mandatory.<br>"Target missing" – The "Target" field is mandatory.<br>"Number is invalid" – The "Target" format is invalid or missing. | String  |

## Voicenter Response Example

```
1  [
2  ]
```

# AddCallsBulk

This method allows you to add multiple destinations to a campaign.

### URI

https://api.voicenter.com/ForwardDialer/Dialer/AddCallsBulk

### Limits

Up to 100,000 destinations can be sent in a single request.  
When sending up to 3,000 destinations, the response message will contain a detailed output for each destination that was sent(like in the response message below). Beyond this amount, a general response answer will be sent.

## Acceptable Request Types

1. POST-JSON

## Possible Response Formats

### 1. POST-JSON

## Client Request Parameters

| Field name      | Description   | Type    | Remarks   |
|-----------------|---|---------|-----------|
| Campaign        | Campaign unique identifier, by GetCampaignList (7.1) method.  | String  | Mandatory |
| Target          | Call target destination phone number.<br>Must be sent with the international prefix (for Israel destinations not mandatory).  | String  | Mandatory |
| CustomerName    | Call target destination name.   | String  | Optional  |
| CallerID        | The phone number that will be display to the destination.<br>Must be phone number in your Voicenter account.  | String  | Optional  |
| Priority        | Call target with higher priority values will dial before lower ones.  | Integer | Optional  |
| CustomData      | You may set your own fields and values that will be associated with each call target destination.<br>This data can be used for pop up and call log.   | Array   | Optional  |
| OriginateTime   | You can set calls to be dial at future date. EPOCH time format.<br>In case this field is not sent in the request, the call will dial as soon as possible.<br>Must be sent together with the "IsDateLocal" field.  | Integer | Optional  |
| IsDateLocal     | Whether the future call date will be in local time as set in the Voicenter Cpanel.<br>In case "false" value was set, the call will be dialed at GMT 0.<br>It is recommended to send "true" value.<br>Must be sent together with the "OriginateTime" field.  | Boolean | Optional  |
| IgnoreDncStatus | In case you activated Voicenter Do-Not-Call-Me service, a service that checks whether to allow calling Israeli destinations, by adding this field to your Auto-Dialer service requests, you will be allowed to make calls to restricted destinations.   | Boolean | Optional  |
| async           | We return a detailed response answer for each destination sent up to an amount of 3,000 destinations per call. Beyond the aforementioned amount, the service returns a general response and does not specify. If a true value is passed in this field, the system will not allow sending an amount exceeding 3,000 destinations. In other words, the service is required to return a detailed response. | Boolean | Optional  |

## Client Request Example (POST - JSON)

```
1  [
2    {
3      "Campaign": "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
```

```

4      "Target": "0501234567",
5      "CustomerName": "John Doe",
6      "CallerID": "0722776772",
7      "Priority": 42,
8      "OriginateTime": 1602750000,
9      "IsDateLocal": "true",
10     "IgnoreDncStatus": "true",
11     "CustomData": {
12         "var_LeadID": 1234567,
13         "var_LeadCampaign": "Facebook"
14     }
15 }
16 {
17     "Campaign": "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
18     "Target": "0501234568"
19 }
20 ]

```

## Voicenter Response Parameters

| Field name  | Description  | Type    |
|-------------|--|---------|
| AddResult   | A list of all the sent calls destinations in the client request and their upload confirmation results.   | Array   |
| Target      | The sent call destination in the client request.   | String  |
| ErrorCode   | Response status:<br>"0" – No errors.<br>"1" – Invalid campaign code.<br>"2" – Target missing.<br>"-2" – Number is invalid.   | Integer |
| Description | Response status description:<br>"OK" – No errors.<br>"Invalid campaign code" – The sent campaign identifier is invalid.<br>"Campaign code missing" - The "Campaign" field is mandatory.<br>"Target missing" – The "Target" field is mandatory.<br>"Number is invalid" – The "Target" format is invalid or missing. | String  |
| CustomData  | The sent costume data in the client request.<br>JSON array.  | Array   |

## Voicenter Response Example

```

1  {
2      "AddResult": [
3          {
4              "Target": "0501234567",
5              "ErrorCode": 0,
6              "Description": "OK",
7              "CustomData": {
8                  "var_LeadID": 1234567,
9                  "var_LeadCampaign": "Facebook"
10             }
11         }
12     ]

```





## Voicenter Response Parameters

| Field name  | Description   | Type    |
|-------------|---|---------|
| ErrorCode   | Response status: "0" – No errors.<br>"1" – Invalid campaign code.<br>"2" – Target missing.  | Integer |
| Description | Response status description: "OK" – No errors.<br>"Invalid campaign code" – The sent campaign identifier is invalid.<br>"Campaign code missing" - The "Campaign" field is mandatory.<br>"Target missing" – The "Target" field is mandatory. | String  |

## Voicenter Response Example

```
1  {  
2    "ErrorCode":0,  
3    "Description":"OK"  
4  }
```

### \*Important.

In case the "Target" value that was sent in the client request does not exist in the campaign, the response will still be "OK". It will just not remove any call destination.

In case the campaign has duplicate calls destinations, it will remove all duplicates targets.

## ClearCampaignCalls

This method removes all calls destination from a campaign.

Totally clear a campaign from all its records.

## URI

<https://api.voicenter.com/ForwardDialer/Dialer/ClearCampaignCalls>

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

- 1.POST-JSON

## Client Request Parameters

| Field name | Description | Type | Remarks |
|------------|-------------|------|---------|
|------------|-------------|------|---------|

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| Campaign   | Campaign unique identifier, provided by GetCampaignList (7.1) method. | String | Mandatory |

### Client Request Example (GET)

```
1 | https://api.voicenter.com/ForwardDialer/Dialer/ClearCampaignCalls?Campaign=XXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

### Client Request Example (POST - JSON)

```
1 | {
2 |   "Campaign": "XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
3 | }
```

### Voicenter Response Parameters

| Field name  | Description  | Integer |
|-------------|--|---------|
| ErrorCode   | Response status:<br>"0" – No errors.<br>"4" – Invalid campaign code.   | Integer |
| Description | Response status description:<br>"OK" – No errors.<br>"Internal error occurred." – The sent campaign identifier is invalid. | String  |

### Voicenter Response Example

```
1 | {
2 |   "ErrorCode": 0,
3 |   "Description": "OK"
4 | }
```

## GetMembersList

Returns all the representatives that were listed to make calls in a campaign.  
This method is only relevant for campaign type – "Agent Dialer".

### URI

https://api.voicenter.com/ForwardDialer/Dialer/GetMembersList

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

- 1.POST-JSON

## Client Request Parameters

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| Campaign   | Campaign unique identifier, provided by GetCampaignList (7.1) method. | String | Mandatory |

## Client Request Example (GET)

```
1 | https://api.voicenter.com/ForwardDialer/Dialer/GetMembersList?Campaign=XXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

## Client Request Example (POST - JSON)

```
1 | {
2 |   "Campaign": "XXXXXXXXXXXXXXXXXXXX"
3 | }
```

## Voicenter Response Parameters

| Field name  | Description   | Type    |
|-------------|---|---------|
| ErrorCode   | Response status:<br>"0" – No errors.                          | Integer |
| Description | Response status description:<br>"OK" – No errors.             | String  |
| Data        | JSON array type.<br>List of the members in a campaign.        | Array   |
| DisplayName | Voicenter extension name.                                     | String  |
| Member      | Voicenter extension unique identifier.<br>Extension SIP code. | String  |

## Voicenter Response Example

```
|
```

```
1  {
2    "Data":[
3      {
4        "Member":"SIPSIP1",
5        "DisplayName":"Walter Melon"
6      }
7      {
8        "Member":"SIPSIP2",
9        "DisplayName":"John Doe"
10     }
11   ]
12   "ErrorCode":0,
13   "Description":"OK"
14 }
```

# AddMember

This method allows you to add an agent to a campaign.  
This method is only relevant for campaign type – "Agent Dialer".

## URI

https://api.voicenter.com/ForwardDialer/Dialer/AddMember

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

- 1.POST-JSON

## Client Request Parameters

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| Campaign   | Campaign unique identifier, provided by GetCampaignList (7.1) method. | String | Mandatory |
| Member     | Voicenter extension unique identifier.<br>Extension SIP code.         | String | Mandatory |

## Client Request Example (GET)

```
1  https://api.voicenter.com/ForwardDialer/Dialer/AddMember?campaign=XXXXXXXXXXXXXXXXXXXXXXXXXXXX
2  &member=SIPSIP1
```

## Client Request Example (POST - JSON)

```
1 {  
2   "Campaign": "XXXXXXXXXX",  
3   "Member": "SIP1SIP1"  
4 }
```

## Voicenter Response Parameters

| Field name  | Description   | Type    |
|-------------|---|---------|
| ErrorCode   | Response status:<br>"0" – No errors.<br>"2" – An issue with the Member value that was sent..  | Integer |
| Description | Response status description:<br>"OK" – No errors.<br>"Member Missing" – The "Member" field is mandatory.                                    | String  |
| Data        | JSON array type.  | Array   |
| TotalAdded  | The amount of added members to a campaign.<br>In case, the value is "-1" – the action was not successful.<br>The sent member was not added. | Integer |

## Voicenter Response Example

```
1 {  
2   "Data": {  
3     "TotalAdded": 2  
4   }  
5   "ErrorCode": 0,  
6   "Description": "OK"  
7 }
```

# RemoveMember

This method allows you to remove an agent from a campaign.  
This method is only relevant for campaign type – "Agent Dialer".

## URI

https://api.voicenter.com/ForwardDialer/Dialer/RemoveMember

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

- 1.POST-JSON

## Client Request Parameters

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| Campaign   | Campaign unique identifier, provided by GetCampaignList (7.1) method. | String | Mandatory |
| Member     | Voicenter extension unique identifier.<br>Extension SIP code.         | String | Mandatory |

## Client Request Example (GET)

```
1 | https://api.voicenter.com/ForwardDialer/Dialer/RemoveMember?campaign=XXXXXXXXXXXXXXXXXXXXXXXXXXXX
2 | &member=SIPSIP1
```

## Client Request Example (POST - JSON)

```
1 | {
2 |   "Campaign": "XXXXXXXXXXXX",
3 |   "Member": "SIP1SIP1"
4 | }
```

## Voicenter Response Parameters

| Field name   | Description  | Type    |
|--------------|--|---------|
| ErrorCode    | Response status:<br>"0" – No errors.<br>"2" – An issue with the Member value that was sent.  | Integer |
| Description  | Response status description:<br>"OK" – No errors.<br>"Member Missing" – The "Member" field is mandatory.   | String  |
| Data         | JSON array type.   | Array   |
| TotalRemoved | The amount of removed members from a campaign.<br>In case, the value is "0" – the action was not successful.<br>The sent member was not removed. | Integer |

## Voicenter Response Example

```
1 | {
2 |   "Data": {
```

```
3      "TotalRemoved":2
4    }
5    "ErrorCode":0,
6    "Description":"OK"
7  }
```

# StopCampaign

Pausing an active campaign from dialing to it's destinations.

## URI

https://api.voicenter.com/ForwardDialer/Dialer/StopCampaign

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

- 1.POST-JSON

## Client Request Parameters

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| Campaign   | Campaign unique identifier, provided by Voicenter or by GetCampaignList (7.1) method. | String | Mandatory |

## Client Request Example (GET)

```
1 | https://api.voicenter.com/ForwardDialer/Dialer/StopCampaign?campaign=XXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

## Client Request Example (POST - JSON)

```
1  {
2    "Campaign": "XXXXXXXXXX"
3  }
```

## Voicenter Response Parameters

| Field name | Description | Type |
|------------|-------------|------|
|            |             |      |



| Field name  | Description  | Type    |
|-------------|--|---------|
| ErrorCode   | Response status:<br>"0" – No errors.<br>"1" – An issue with the Campaign value that was sent.                                  | Integer |
| Description | Response status description:<br>"OK" – No errors.<br>"Invalid campaign code" – An issue with the Campaign value that was sent. | String  |

## Voicenter Response Example

```
1  {
2    "ErrorCode":1,
3    "Description":"Invalid campaign code"
4  }
```

# StartCampaign

If a campaign is currently paused, this action will resume it to dial to the destinations.

## URI

https://api.voicenter.com/ForwardDialer/Dialer/StartCampaign

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

- 1.POST-JSON

## Client Request Parameters

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| Campaign   | Campaign unique identifier, provided by GetCampaignList (7.1) method. | String | Mandatory |

## Client Request Example (GET)

```
1 | https://api.voicenter.com/ForwardDialer/Dialer/StartCampaign?campaign=XXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

## Client Request Example (POST - JSON)

```

1  {
2    "Campaign": "XXXXXXXXXXXX"
3  }

```

## Voicenter Response Parameters

| Field name  | Description  | Type    |
|-------------|--|---------|
| ErrorCode   | Response status:<br>"0" – No errors.<br>"1" – An issue with the Campaign value that was sent.                                  | Integer |
| Description | Response status description:<br>"OK" – No errors.<br>"Invalid campaign code" – An issue with the Campaign value that was sent. | String  |

## Voicenter Response Example

```

1  {
2    "ErrorCode":1,
3    "Description":"Invalid campaign code"
4  }

```

# GetCampaignPendingCalls

Returns a list of all waiting to be dialed calls destinations.

## URI

<https://api.voicenter.com/ForwardDialer/Dialer/GetCampaignPendingCalls>

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

- 1.JSON

## Client Request Parameters

| Field name | Description                                 | Type   | Remarks   |
|------------|---|--------|-----------|
| Campaign   | Personal identifier, provided by Voicenter. | String | Mandatory |

## Client Request Example (GET)

```
1 | https://api.voicenter.com/ForwardDialer/Dialer/GetCampaignPendingCalls?Campaign=XXXXXXXXXXXXXXXXXXXXX
```

## Client Request Example (POST - JSON)

```
1 | {
2 |   "Campaign": "XXXXXXXXXXXXXXXXXXXXX"
3 | }
```

## Voicenter Response Parameters

| Field name        | Description   | Type    |
|-------------------|---|---------|
| ErrorCode         | Response status:<br>"0" – No errors.<br>"2" – An issue with the Campaign value that was sent.                                   | Integer |
| Description       | Response status description:<br>"OK" – No errors.<br>"Internal error occured" – An issue with the Campaign value that was sent. | String  |
| Data              | JSON array type.<br>Contains:<br>1. Campaign – General data.<br>2. Calls - List of the waited to be dialed calls                | Array   |
| Campaign          | JSON array type.<br>The campaign general data.  | Array   |
| Name              | The Voicenter campaign name.  | String  |
| StatusName        | Campaign status:<br>"Enabled" – Active campaign.<br>"Disabled" – Stopped campaign.  | String  |
| TotalPendingCalls | The sum of waiting records destinations to be dial.   | Integer |
| MaxPriority       | Returns the highest priority value that was set for a call destination record in the campaign.                                  | Integer |
| MinPriority       | Returns the lowest priority value that was set for a call destination record in the campaign.                                   | Integer |

| Field name         | Description  | Type    |
|--------------------|--|---------|
| TotalAwaitingCalls | Relevant only for IVR dialer campaigns.<br>The sum of waiting records destinations to be dial at the IVR dialer service.<br>This field is more recommended to use then "TotalPendingCalls" for IVR dialer campaign.<br>In case of agent dialer campaign the returned value will be "null". | Integer |
| Code               | The campaign unique identifier code.<br>Needed for the auto dialer API methods.  | String  |
| Calls              | JSON array type.<br>List of the waited to be dialed calls.   | Array   |
| Phone              | Call target destination phone number.<br>Must be sent with the international prefix (for Israel destinations not mandatory).   | String  |
| CustomerName       | Call target destination name.  | String  |
| CallerID           | The phone number that will be display to the destination.<br>Must be phone number in your Voicenter account.<br>This field will display value only if this data was sent via API request.  | String  |
| Priority           | Call target with higher priority values will dial before lower ones.   | Integer |
| CustomData         | You may set your own fields and values that will be associated with each call target destination.<br>This data can be used for pop up and call log.  | Array   |
| OriginateTime      | You can set calls to be dial at future date.<br>EPOCH time format.<br>In case this field is not sent in the request, the call will dial as soon as possible.<br>Must be send together with "IsDateLocal" field.  | Integer |
| IsDateLocal        | Whether the future call date will be in local time as set in the Voicenter Cpanel.<br>In case "false" value was set, the call will be dialed at GMT 0.<br>It is recommended to send "true" value.<br>Must be send together with "OriginateTime" field.                                     | Boolean |
| CallStatus         | JSON array type.<br>The status of the waited to be dialed call.  | Array   |
| Status             | "1" – A call pending to be dial.   | Integer |
| Description        | "Pending" - A call pending to be dial.   | String  |

## Voicenter Response Example

```
1  {
2    "Data":{
3      "Calls":[
4        {
5          "Phone": "972525599100",
6          "CallerID": "",
7          "CallerName": "",
```

```

8      "OriginateTime":1602819000,
9      "Priority":42,
10     "CustomerName":"John Doe",
11     "CallStatus":{
12         "Status":1,
13         "Description":"Pending"
14     }
15     "CustomData":{
16         "var_test1":"1",
17         "var_test2":"some value"
18     }
19 }
20 {
21     "Phone":"972525599100",
22     "CallerID":"",
23     "CallerName":"",
24     "OriginateTime":1602755000,
25     "Priority":42,
26     "CustomerName":"John Doe",
27     "CallStatus":{
28         "Status":2,
29         "Description":"Originated"
30     }
31     "CustomData":{
32         "var_test1":"1",
33         "var_test2":"some value"
34     }
35 }
36 ]
37 "Campaign":{
38     "Name":"Campaign2",
39     "StatusName":"Enabled",
40     "TotalPendingCalls":4,
41     "TotalOriginatedCalls":0,
42     "MaxPriority":42,
43     "MinPriority":42,
44     "TotalAwaitingCalls":-1,
45     "Code":"jngqka31F7n1ujPZb0eEJsMzfVaoU2jp"
46 }
47 }
48 "ErrorCode":0,
49 "Description":"OK"
50 }

```

# UpdateCall

Will update an existing call target destination.  
 \*In case call target does not exist in the campaign, it will add it (AddCall method 6.2).

## URI

https://api.voicenter.com/ForwardDialer/Dialer/UdpateCall

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

Client Request Parameters

| Field name    | Description  | Type    | Remarks   |
|---------------|--|---------|-----------|
| Campaign      | Campaign unique identifier, provided by GetCampaignList (7.1) method.  | String  | Mandatory |
| Target        | Call target destination phone number.<br>Must be sent with the international prefix (for Israel destinations not mandatory).   | String  | Mandatory |
| CustomerName  | Call target destination name.  | String  | Optional  |
| CallerID      | The phone number that will be display to the destination.<br>Must be phone number in your Voicenter account.   | String  | Optional  |
| Priority      | Call target with higher priority values will dial before lower ones.   | Integer | Optional  |
| CustomData    | You may set your own fields and values that will be associated with each call target destination.<br>This data can be used for pop up and call log.  | Array   | Optional  |
| OriginateTime | You can set calls to be dial at future date.<br>EPOCH time format.<br>In case this field is not sent in the request, the call will dial as soon as possible.<br>Must be send together with "IsDateLocal" field.  | Integer | Optional  |
| IsDateLocal   | Whether the future call date will be in local time as set in the Voicenter Cpanel.<br>In case "false" value was set, the call will be dialed at GMT 0.<br>It is recommended to send "true" value.<br>Must be send together with "OriginateTime" field. | Boolean | Integer   |

Client Request Example (GET)

```
1 https://api.voicenter.com/ForwardDialer/Dialer/UpdateCall?Campaign=XXXXXXXXXXXXXXXXXXXX&Target=0501234567&CustomerName=John Doe
```

Client Request Example (POST - JSON)

```
1 {
2   "Campaign": "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
3   "Target": "0501234567",
4   "CustomerName": "John Doe",
5   "CallerID": "0722776772",
6   "Priority": 42,
7   "OriginateTime": 1602750000,
8   "IsDateLocal": "true",
```

```
9      "CustomData": {
10        "var_LeadID": 1234567,
11        "var_LeadCampaign": "Facebook"
12      }
13    }
```

## Voicenter Response Parameters

| Field name  | Description   | Integer |
|-------------|---|---------|
| ErrorCode   | Response status: "0" – No errors.<br>"1" – Invalid campaign code.<br>"2" – Target missing.<br>"-2" – Number is invalid.   | Integer |
| Description | Response status description: "OK" – No errors.<br>"Invalid campaign code" – The sent campaign identifier is invalid.<br>"Campaign code missing" - The "Campaign" field is mandatory.<br>"Target missing" – The "Target" field is mandatory.<br>"Number is invalid" – The "Target" format is invalid or missing. | String  |

## Voicenter Response Example

```
1  {
2    "ErrorCode": 0,
3    "Description": "OK"
4  }
```

# GetCampaignDetails

Returns details for requested campaign.

## URI

https://api.voicenter.com/ForwardDialer/Dialer/GetCampaignDetails

## Acceptable Request Types

- 1.POST-JSON
- 2.GET

## Possible Response Formats

- 1.POST-JSON

## Client Request Parameters

| Field name | Description | Type | Remarks |
|------------|-------------|------|---------|
|            |             |      |         |

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| Campaign   | Campaign unique identifier, provided by GetCampaignList (7.1) method. | String | Mandatory |

Client Request Example (GET)

```
1 | https://api.voicenter.com/ForwardDialer/Dialer/GetCampaignDetails?Campaign=XXXXXXXXXXXXXXXXXXXXX
```

Client Request Example (POST - JSON)

```
1 | {
2 |   "Campaign": "XXXXXXXXXXXXXXXXXXXXX"
3 | }
```

Voicenter Response Parameters

| Field name         | Description  | Type    |
|--------------------|--|---------|
| ErrorCode          | Response status:<br>"0" – No errors.   | Integer |
| Description        | Response status description:<br>"OK" – No errors.  | String  |
| Data               | JSON array type.<br>List of the requested automatic dialer campaigns.  | Array   |
| Name               | The Voicenter campaign name.   | String  |
| StatusName         | Campaign status:<br>"Enabled" – Active campaign.<br>"Disabled" – Stopped campaign.   | String  |
| TotalPendingCalls  | The sum of waiting records destinations to be dial.  | Integer |
| MaxPriority        | Returns the highest priority value that was set for a call destination record in the campaign.   | Integer |
| MinPriority        | Returns the lowest priority value that was set for a call destination record in the campaign.  | Integer |
| TotalAwaitingCalls | Relevant only for IVR dialer campaigns.<br>The sum of waiting records destinations to be dial at the IVR dialer service.<br>This field is more recommended to use then "TotalPendingCalls" for IVR dialer campaign.<br>In case of agent dialer campaign the returned value will be "-1". | Integer |
|                    |  |         |



| Field name | Description   | Type   |
|------------|---|--------|
| Code       | The campaign unique identifier code.<br>Needed for the auto dialer API methods. | String |

## Voicenter Response Example

```
1  {
2    "Data":{
3      "Name":"Campaign1",
4      "StatusName":"Enabled",
5      "TotalPendingCalls":23,
6      "MaxPriority":1,
7      "MinPriority":1,
8      "TotalAwaitingCalls":23,
9      "Code":"XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
10   }
11   "ErrorCode":0,
12   "Description":"OK"
13 }
```

| Field name             | Description   | Type    | Remarks   |
|------------------------|---|---------|-----------|
| Campaign               | Campaign unique identifier, provided by GetCampaignList (7.1) method.                     | String  | Mandatory |
| setQueueForMonitoring  | Queue unique identifier.<br>Can be found in the "Queues" section in the Cpanel.           | Integer | Mandatory |
| setQueueMaximumCallers | Maximum amount of waiting calls in the selected queue.<br>Value must be greater than "0". | Integer | Mandatory |
| setCoefficient         | The campaign dialing call rate coefficient value.<br>Value must be greater than "1".      | Integer | Mandatory |

## Client Request Example (GET)

```
1  https://api.voicenter.com/ForwardDialer/Dialer/UpdateCampaignStrategy?Campaign=XXXXXXXXXXXXXXXXXXXX
2  &setQueueForMonitoring=123456789&setQueueMaximumCallers=1&setCoefficient=2
```

## Client Request Example (POST - JSON)

```
1  {
2    "Campaign":"xxxxxxxxxxxxxxxxxxxx",
3    "setQueueForMonitoring":123456789,
4    "setQueueMaximumCallers":1,
5    "setCoefficient":2
6  }
```

## Voicenter Response Parameters

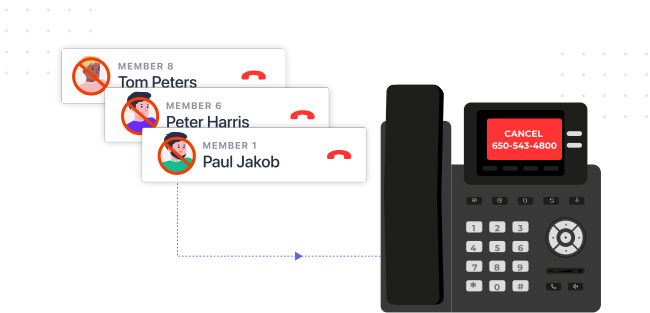
| Field name  | Description                                       | Type    |
|-------------|---|---------|
| ErrorCode   | Response status:<br>"0" – No errors.              | Integer |
| Description | Response status description:<br>"OK" – No errors. | String  |

## Voicenter Response Example

```
1 {
2   "ErrorCode":0,
3   "Description":"OK"
4 }
```

# Blacklist API

The blacklist API can be used to add or remove phone numbers from the blacklist service via simple WEB request.



## AddBlackList

### General Explanation

This method will allow you to easily add a destination (phonenumber) to you Voicenter organizational Blacklist.

### URI

```
1 | https://api.voicenter.com/Blacklist/AddBlackList
```

### Acceptable Request Types

1. POST-JSON
2. GET

### Possible Response Formats

1. POST-JSON

### Client Request Parameters

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| Code       | Personal identifier, provided by Voicenter.   | String | Mandatory |
| Phones     | JSON array type.<br>List of the requested destination to be added to the blacklist service. | Array  | Mandatory |
| Phone      | The blocked destination phone number.   | String | Mandatory |
| Name       | The name of the blocked call destination.<br>Supported only in POST requests.               | String | Optional  |

## Client Request Example (GET)

```
1 | https://api.voicenter.com/Blacklist/AddBlackList?code=xxxxxxxxxxxxxxxxxxxx&phones=0501234567&phones=031234567
```

## Client Request Example (POST - JSON)

```
1 | {
2 |   "Code": "XXXXXXXXXXXXXXXXXXXX",
3 |   "Phones": [
4 |     {
5 |       "Phone": "0501234567",
6 |       "Name": "John doe"
7 |     }
8 |     {
9 |       "Phone": "031234567",
10 |      "Name": "Walter Melon"
11 |     }
12 |   ]
13 | }
```

## Voicenter Response Parameters

| Field name   | Description   | Type    |
|--------------|---|---------|
| ErrorCode    | General response status: "0" – No errors.<br>"1" – The Code value or format is invalid.<br>"2" – The Phone field is missing or invalid.   | Integer |
| ErrorMessage | General response status description: "OK" – No errors.<br>"Authorization failed" – The Code value or format is invalid.<br>"The Phone field is missing or invalid" - The "Phone" field is mandatory.  | String  |
| Phones       | JSON array type.<br>For each phone destination sent to the blacklist service, the corresponding result of the attempt to add it to the service is returned.   | Array   |
| ErrorCode    | General response status: "0" – No errors.<br>"1" – The sent phone destination is not in a valid format.<br>Please send phone destination in the international standard format – E.164.<br>"2" – There was an internal error when trying to add the specific destination.<br>Please contact Voicenter support team.  | Integer |
| ErrorMessage | Response status description to each sent destination in the request:<br>"OK" – No errors.<br>"Please make sure the phone destination is in a valid format" - The sent phone destination is not in a valid format.<br>Please send phone destination in the international standard format – E.164.<br>"Internal Error" – There was an internal error when trying to add the specific destination.<br>Please contact Voicenter support team. | String  |

## Voicenter Response Example

```
1  {
2    "ErrorCode":0,
3    "ErrorMessage": "OK",
4    "Phones": [
5      {
6        "ErrorCode":0,
7        "ErrorMessage": "OK",
8        "Phone": "972501234567"
9      }
10     {
11       "ErrorCode":0,
12       "ErrorMessage": "OK",
13       "Phone": "97231234567"
14     }
15   ]
16 }
```

## RemoveBulkFromBlacklist

### General Explanation

A method for removing phone numbers from the organization blacklist service.

URI

<https://api.voicenter.com/Blacklist/RemoveBulkFromBlacklist>

### Acceptable Request Types

- 1. POST-JSON
- 2. GET

### Possible Response Formats

- 1. POST-JSON

### Client Request Parameters

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| Code       | Personal identifier, provided by Voicenter.   | String | Mandatory |
| Phones     | JSON array type.<br>List of the requested destination to be added to the blacklist service. | Array  | Mandatory |
| Phone      | The blocked destination phone number.   | String | Mandatory |

### Client Request Example (GET)

```
1 https://api.voicenter.com/Blacklist/RemoveBulkFromBlacklist?code=xxxxxxxxxxxxxxxxxxx&phones=0501234567&phones=031
```

#### Client Request Example (POST - JSON)

```
1 {  
2   "Code": "XXXXXXXXXXXXXXXXXX",  
3   "Phones": [  
4     {  
5       "Phone": "0501234567"  
6     }  
7     {  
8       "Phone": "031234567"  
9     }  
10  ]  
11 }
```

## Voicenter Response Parameters

| Field name   | Description   | Type    |
|--------------|---|---------|
| ErrorCode    | General response status:<br>"0" – No errors.<br>"1" – The Code value or format is invalid.<br>"2" – The Phone field is missing or invalid.  | Integer |
| ErrorMessage | General response status description:<br>"OK" – No errors.<br>"Authorization failed" – The Code value or format is invalid.<br>"The Phone field is missing or invalid" - The "Phone" field is mandatory.   | String  |
| Phones       | JSON array type.<br>For each phone destination sent to the blacklist service, the corresponding result of the attempt to add it to the service is returned.   | Array   |
| ErrorCode    | General response status:<br>"0" – No errors.<br>"1" – The sent phone destination is not in a valid format.<br>Please send phone destination in the international standard format – E.164.<br>"2" – There was an internal error when trying to add the specific destination.<br>Please contact Voicenter support team.   | Integer |
| ErrorMessage | Response status description to each sent destination in the request:<br>"OK" – No errors.<br>"Please make sure the phone destination is in a valid format" - The sent phone destination is not in a valid format.<br>Please send phone destination in the international standard format – E.164.<br>"Internal Error" – There was an internal error when trying to add the specific destination.<br>Please contact Voicenter support team. | String  |

## Voicenter Response Example

```
1  {
2    "ErrorCode":0,
3    "ErrorMessage":"OK",
4    "Phones":[
5      {
6        "ErrorCode":0,
7        "ErrorMessage":"OK",
8        "Phone":"972501234567"
9      }
10     {
11       "ErrorCode":0,
12       "ErrorMessage":"OK",
13       "Phone":"97231234567"
14     }
15   ]
16 }
```

## Login/Logout API

Voicenter Login/Logout API allows you to manage your representatives' online statuses in the shift in your internal system. A representative can easily update his/her status (Login/Logout/Lunch and etc.) by a simple web request.



## General Explanation

**URI**



```
1 | https://api.voicenter.com/UserLogin/SetStatusFromAPI
```

## Acceptable Request Types

- 1. POST-JSON
- 2. GET

## Possible Response Formats

- 1. POST-JSON

## Client Request Parameters

| Field name    | Description  | Type    | Remarks   |
|---------------|--|---------|-----------|
| Code          | Personal identifier, provided by Voicenter.  | String  | Mandatory |
| UserId        | The Voicenter user identifier, provided by Voicenter.  | String  | Mandatory |
| ExtensionUser | The Voicenter extension SIP code, provided by Voicenter.   | String  | Mandatory |
| Status        | The user online work status:<br>"1" - Login<br>"2" - Logout<br>"3" - Lunch<br>"5" - Administrative<br>"7" - Private<br>"9" - Other<br>"11" - Training<br>"12" - Team meeting<br>"13" - Brief<br>"The names of these statuses can be changed in the CPanel. | Integer | Mandatory |

## Client Request Example (GET)

```
1 | https://api.voicenter.com/UserLogin/SetStatusFromAPI?Code=XXXXXXXXXX&UserId=123456789
2 | &ExtensionUser=sipsip&Status=1
```

## Client Request Example (POST - JSON)

```
1 | {
2 |   "Code": "XXXXXXXXXX",
3 |   "UserId": "123456789",
4 |   "ExtensionUser": "sipsip",
5 |   "Status": 1
6 | }
```

## Voicenter Response Parameters

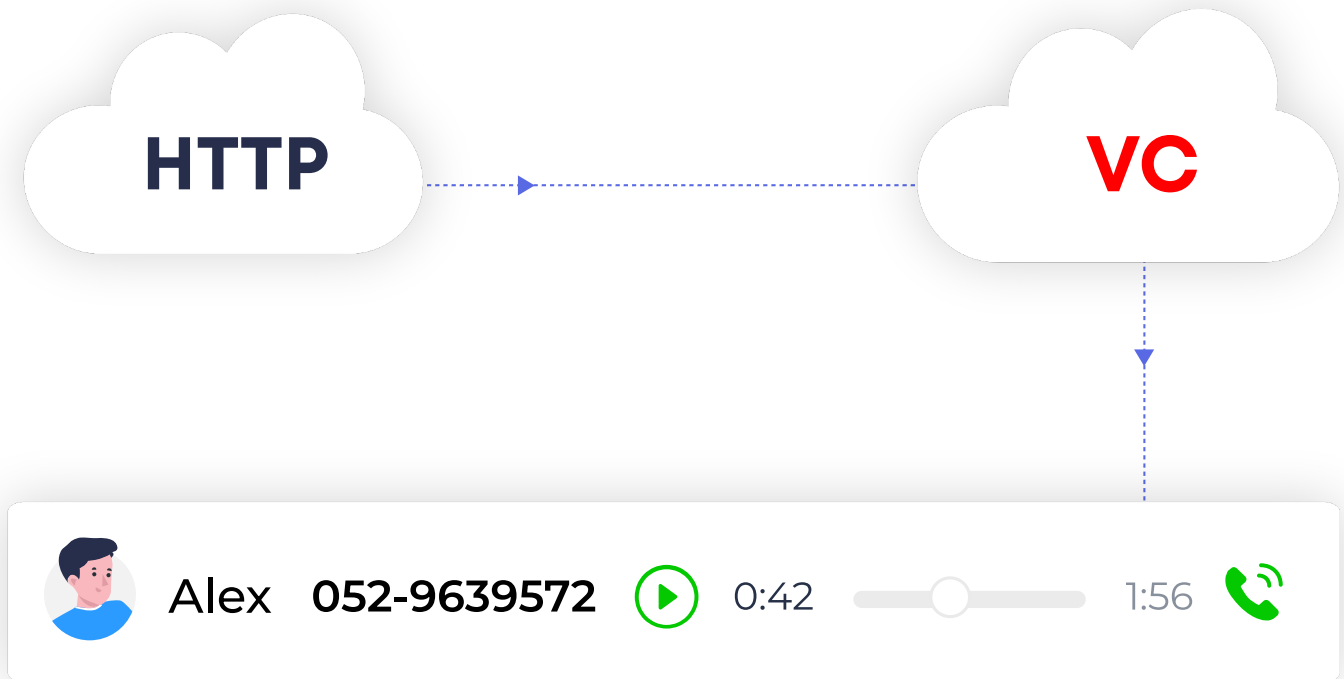
| Field name        | Description  | Type    |
|-------------------|--|---------|
| Status            | Response status:<br>"1" – The request was successful, No errors.<br>"2" – The Code or User ID value is invalid.<br>"3" – The Extension value is invalid.<br>"4" – Status not supported or the ExtensionUser value is invalid.  | Integer |
| ServerError       | "0" – There was something wrong in the request format.<br>"1" – The request was successful.<br>"4" – Internal error.<br>"6" – Missing parameters.  | Integer |
| StatusErroMessage | Response status description:<br>"OK" – The request was successful, No errors.<br>"Authorization Failed" - The Code value or format is invalid.<br>"Phone error." – The "Phone" field is mandatory.<br>"Status not support" – The status field value is not valid.<br>"Extension is already in use" – If a user tries to connect to an extension that another user is connected to it.<br>"No extension found" – The Extension User value is not valid.<br>"Error on update" – There was an error on the server side. | String  |

## Voicenter Response Example

```
1  {
2    "Status":1,
3    "ServerError":1,
4    "StatusErroMessage":"OK"
5  }
```

## Active Calls API

Active calls API allows you to get a "snapshot" of your call center current telephony activity, by using a simple WEB request. You can also monitor your users and queues activity in any given moment.



## General Explanation

The active calls API gives you a clear picture of how many active calls your organization currently have and/or how many calling clients are currently waiting in your queue/s.

You can also perform your own dedicated popup with this api when an agent answer and call they press a button in your crm system which triggers an api request to Voicenter Active Calls api.

## GetExtensionsCalls

Returns data by a specific extension or all extensions in your call center.

### URI

1 | <https://monapisec.voicenter.co.il/comet/API/GetExtensionsCalls>

## Acceptable Request Types

- 1. POST-JSON
- 2. GET

## Possible Response Formats

- 1.POST-JSON

## Client Request Parameters

| Field name | Description  | Type   | Remarks   |
|------------|--|--------|-----------|
| code       | Personal identifier, provided by Voicenter.  | String | Mandatory |
| extension  | In case you wish to get result for a specific extension.<br>Extension SIP identifier, provided by Voicenter.<br>If this field is not sent in the request, the API will return all extensions data. | String | Optional  |

## Client Request Example (GET)

```
1 | https://monapisec.voicenter.co.il/comet/API/GetExtensionsCalls?code=XXXXXXXXXXXX&extension=SIPSIP
```

## Client Request Example (POST - JSON)

```
1 | {
2 |   "code": "XXXXXXXXXXXX",
3 |   "extension": "SIPSIP"
4 | }
```

## Voicenter Response Parameters

| Field name | Description   | Type    |
|------------|---|---------|
| ERR        | Response status:<br>"0" – No errors.<br>"1" – The request format is invalid.<br>"2" – Either the sent parameters values are incorrect or Voicenter internal issue.  | Integer |
| DESC       | Response status description:<br>"OK" – No errors.<br>"Authorization failed." - The request format is invalid.<br>"Internal error." – Either the sent parameters values are incorrect or Voicenter internal issue. | String  |

| Field name       | Description  | Type    |
|------------------|--|---------|
| EXTENSIONS       | JSON array type.<br>List of the requested extensions.  | Array   |
| name             | The Voicenter user name that the extension is associated with.   | String  |
| userID           | The Voicenter user ID that the extension is associated with.   | Integer |
| representative   | The Voicenter user name that is currently logged in to the extension.  | String  |
| onlineUserID     | The Voicenter user ID that is currently logged in to the extension.  | Integer |
| username         | The extension unique SIP code identifier.  | String  |
| onlineUserStatus | <p>The user status that is currently logged into the extension:</p> <p>"1" – Login.</p> <p>"2" – Logout. *</p> <p>"3" – Lunch.</p> <p>"5" – Administrative.</p> <p>"7" – Private.</p> <p>"9" – Other.</p> <p>"11" – Training.</p> <p>"12" - Team meeting.</p> <p>"13" – Brief.</p> <p>*Logout is the default status that is return in case there is no logged in user on an extension.</p> | Integer |
| calls            | JSON array type.<br>Displays the currently active calls at the extension.  | Array   |
| callStarted      | The time that the call started in EPOCH time.  | Integer |
| callAnswered     | The time that the call was answered in EPOCH time.<br>In case the call was not answered yet, it will return "0".   | Integer |
| answered         | <p>Was the call answered?</p> <p>"0" – No (or not yet).</p> <p>"1" – Yes.</p>  | Integer |
| callername       | Caller name that is shown at the destination phone (Supported only for Voicenter extensions).  | String  |
| callerphone      | Caller caller ID.  | String  |
| callstatus       | <p>The call status:</p> <p>"Ringing" – An incoming call is currently ringing at the extension.</p> <p>"Dialing" – In case the extension is making an outgoing call and the call was not answered yet.</p> <p>"Talking" – There is an ongoing conversation at the extension (an incoming or outgoing call).</p> <p>"Hold" – In case the call was put on "Hold".</p>                         | String  |
|                  |  |         |

| Field name | Description  | Type    |
|------------|--|---------|
| customdata | JSON array type.<br>Displays custom data that was sent to Voicenter.<br>In addition, if a specific call has an origin call it will display here with the field name "OriginalIvrUniqueID".<br>For example: if a call was first answered by one representative who then transferred it to another representative.<br>The second part of the call that was transferred, will have in the "CustomData" field a ivrid value of the original call, thus associating both calls. | Array   |
| direction  | General call direction:<br>"Incoming".<br>"Outgoing".  | String  |
| ivrid      | Displays the ID code of the specific call.<br>Unique identifier.   | String  |
| recording  | JSON array type.<br>Displays the call recording data.  | Array   |
| Filename   | The recording file name.   | String  |
| IsMuted    | Displays if the current conversation is now recorded?<br>"0" – Not muted. The conversation is now being recorded.<br>"1" – Muted. The conversation is not being recorded now.<br>With Mute Call API you can control which part of a call will be recorded.   | Integer |
| did        | In case of an incoming call, displays the phone number that the caller dialed to.  | String  |

## Voicenter Response Example

```
1  {
2    "ERR":0,
3    "DESC":"OK",
4    "EXTENSIONS":[
5      {
6        "name":"User 1",
7        "representative":"User 1",
8        "username":"SIPSIP1",
9        "extensionID":875756567,
10       "userID":6946792,
11       "onlineUserID":0,
12       "onlineUserStatus":2,
13       "calls":[
14         ]
15     }
16     {
17       "name":"User 2",
18       "representative":"User 2",
19       "username":"SIPSIP2",
20       "extensionID":97483478,
21       "userID":46454322,
22       "onlineUserID":46454322,
23       "onlineUserStatus":1,
24       "calls":[
25         {
26           "callStarted":1602465818,
```

```
27         "callAnswered":1602465819,
28         "answered":1,
29         "callername":"Voicenter",
30         "callerphone":"0722776772",
31         "callstatus":"Talking",
32         "customdata":{
33             }
34         "direction":"Outgoing",
35         "ivrid":"2020101201cc7b38df",
36         "recording":{
37             "Filename":"2020101201cc7b38df-aws-SIPSIP2-972722776772.mp3",
38             "IsMuted":0
39         }
40         "did":""
41     }
42 ]
43 }
44 ]
45 }
```

# GetQueuesCallers

Returns data by a specific queue or all queues in your call center.

## URI

```
1 | https://monapisec.voicenter.co.il/comet/API/GetQueuesCallers
```

## Acceptable Request Types

- 1. POST - JSON
- 2. GET

## Possible Response Formats

- 1. POST-JSON

## Client Request Parameters

| Field name | Description   | Type   | Remarks   |
|------------|---|--------|-----------|
| code       | Personal identifier, provided by Voicenter.   | String | Mandatory |
| queue      | In case you wish to get result for a specific queue.<br>Queue ID identifier, provided by Voicenter.<br>If this field is not sent in the request, the API will return all queues data. | String | Optional  |

## Client Request Example (GET)

```
|
```

1 | https://monapisec.voicenter.co.il/comet/API/GetQueuesCallers?code=XXXXXXXXXXXX&queue=XXXXXXXX

### Client Request Example (POST - JSON)

```
1 {
2   "code": "XXXXXXXXXXXX",
3   "queue": "XXXXXXXX"
4 }
```

### Voicenter Response Parameters

| Field name | Description   | Type    |
|------------|---|---------|
| ERR        | Response status:<br>"0" – No errors.<br>"1" – The request format is invalid.<br>"2" – Either the sent parameters values are incorrect or Voicenter internal issue.  | Integer |
| DESC       | Response status description:<br>"OK" – No errors.<br>"Authorization failed." - The request format is invalid.<br>"Queue is not found in this account" – In case the queue field value is incorrect.<br>"Internal error." – Either the sent parameters values are incorrect or Voicenter internal issue. | String  |
| QUEUES     | JSON array type.<br>List of the requested queues.   | Array   |
| Name       | The queue name.   | String  |
| ID         | The queue ID.   | Integer |
| Weight     | The current queue weight configuration.   | Integer |
| Callers    | JSON array type.<br>Displays the active calls at the queue.   | Aray    |
| Phone      | Caller caller ID.   | String  |
| CallID     | Displays the ID code of the specific call.<br>Unique identifier.  | String  |
| JoinTime   | The time that the call entered the queue in EPOCH time.   | Integer |
| Duration   | The time that the call is currently waiting in the queue in Seconds.  | Integer |

### Voicenter Response Example

```
1 {
```



```
2  "ERR":0,
3  "DESC":"OK",
4  "QUEUES":[
5    {
6      "Name":"QUEUE 1",
7      "ID":12345678,
8      "Weight":5,
9      "Callers":[
10     {
11       "Phone":"0722776772",
12       "CallID":"202010131430590714966",
13       "JoinTime":1602599565,
14       "Duration":21
15     }
16     {
17       "Phone":"0501234567",
18       "CallID":"202010131433000d14cc4b",
19       "JoinTime":1602599583,
20       "Duration":3
21     }
22   ]
23 }
24 {
25   "Name":"QUEUE 2",
26   "ID":87654321,
27   "Weight":0,
28   "Callers":[
29   ]
30 }
31 ]
32 }
```

# Organizational Extension List

Use Voicenter's Organizational Extension List to view your the active extensions in your organization, this API pairs well with the Login/Logout API since you can give the agent the option of choosing with extension to log into.

## General Explanation

### Uri

1 | <https://monitor.voicenter.co.il/Comet/api/GetExtensions>

### Acceptable Request Types

1. POST-JSON
2. GET

### Possible Response Formats

1. POST-JSON

### Client Request Parameters

| Field name | Description   | Type    | Remarks   |
|------------|---|---------|-----------|
| Code       | Personal identifier, provided by Voicenter.   | String  | Mandatory |
| showAll    | Allows to set if the return extensions list by the organization hierarchy.<br>If you wish to get all the extensions data in the all organization or only for a specific department.<br>If this field is not sent in the request, the API will return all extensions in the organization.<br>"False" or "0" – Returns all extension in a specific department.<br>You will need to receive the "Code" identifier for each department you wish to use.<br>"True" or "1" – Returns all extension in the organization. | Boolean | Optional  |

### Client Request Example (GET)

1 | <https://monitor.voicenter.co.il/Comet/api/GetExtensions?code=XXXXXXXXXXXX&showAll=0>

## Client Request Example (POST - JSON)

```
1 {
2   "code": "XXXXXXXXXXXXX",
3   "showAll": 0
4 }
```

## Voicenter Response Parameters

| Field name  | Description  | Type    |
|-------------|--|---------|
| ERR         | Response status:<br>"0" – No errors.<br>"1" – The sent request Code format is invalid.<br>* In case the sent "Code" value is invalid you will simply get an empty "EXTENSIONS" array response. | Integer |
| DESC        | Response status description:<br>"OK" – No errors.<br>"Unauthorized" - The Code value or format is invalid.   | String  |
| EXTENSIONS  | A list of the requested extensions data.   | Array   |
| SIP         | The extension's personal identifier.<br>The SIP user code.   | String  |
| Name        | The extension's name (as it appear in the Voicenter Cpanel).   | String  |
| SpeedDial   | The extension's personal speed dial.<br>Speed dial – a short phone number for internal communication between the organization extensions.  | String  |
| AccountID   | The department that the extension is set on personal identifier(as it appear in the Voicenter Cpanel).   | Integer |
| AccountName | The department that the extension is set on name (as it appear in the Voicenter Cpanel).   | String  |
| UserName    | The user that the extension is set on name (as it appear in the Voicenter Cpanel).   | String  |
| UserEmail   | The user that the extension is set on email field (as it appear in the Voicenter Cpanel).  | String  |

## Voicenter Response Example

```
1 {
2   "ERR": 0,
3   "DESC": "OK",
4   "EXTENSIONS": [
5     {
6       "SIP": "SIPSIP1",
```

```
7      "Name": "Extension 1",
8      "SpeedDial": "11",
9      "AccountID": 12345678,
10     "AccountName": "Voicenter Account",
11     "UserName": "John Doe",
12     "UserEmail": "John.d@voicenter.co.il"
13   }
14   {
15     "SIP": "SIPSIP2",
16     "Name": "Extension 2",
17     "SpeedDial": "12",
18     "AccountID": 12345679,
19     "AccountName": "Sales Department",
20     "UserName": "Walter Melon",
21     "UserEmail": "Walter.m@voicenter.co.il"
22   }
23 ]
24 }
```

# Mute Call Recording API

---

Mute Call recording API allows clients who use Voicenter call recording services the ability to control in real time, whether a specific part of a call will be muted. It is mostly used when in a specific part of a call, the caller provides the representative restricted information (credit card number etc.) which you prefer not to store in your call recording database.



## Mute by extension

When sending a Mute request by Voicenter extension SIP ID, all the active calls on this extension will be muted (multiple calls can be in case of a conference call, attended transfer calls, etc.).

Important. In case a Mute request by extension was sent, and after that the extension will make a new call, the initial Mute request will not include the new call and another mute request must be sent to mute the new call.

For example, a representative who speaks to a customer and is about to get the customer credit card number.

The representative will send a mute request at this stage so this part of the call will not be recorded.

If after the representative sends the Mute request the customer asks the representative to call another person to get the credit card details, this new call will not be muted, unless the representative sends another Mute call request.

## URI

1 | <https://<monitorX>.voicenter.co/api/MuteUnmuteCalls>

Please notice that <monitorX> is the name of the monitor server assigned to your account.

You can ask Voicenter support for the static monitor server of your specific account, or deploy the RealTimeAPI to get this list dynamically.

<https://www.voicenter.com/API/real-time>

## Acceptable request types

- 1. POST-JSON
- 2. GET

## Possible Response formats

- 1. POST-JSON

## Client Request Parameters

| Field name | Description                                      | Type   | Remarks   |
|------------|--|--------|-----------|
| extension  | SIP ID of the extension, as found in the Cpanel. | String | Mandatory |
| state      | 0 – Unmute .<br>1 - Mute.                        | String | Mandatory |

## Client Request Example (GET)

```
1 | https://YOUR_MONITOR.voicenter.co/api/MuteUnmuteCalls?extension=XXXXXXX&state=1
```

## Client Request Example (POST - JSON)

```
1 | {
2 |   "extension": "API1API",
3 |   "state": "0"
4 | }
```

## Voicenter Response Parameters

| Field name | Description  | Type    |
|------------|--|---------|
| ErrorCode  | Describes the status of the client request.<br>The status 200 will indicate a succesfull request.  | Integer |
| Message    | Response status description.<br>Response status description:<br>"Success" – No errors.<br>"UniquelvrID not found" - The Code value or format is invalid.<br>"Parameters are not valid, should be \"ivrid\" and \"mute\" - The sent parameters are invalid. | String  |

| Field name | Description                       | Type   |
|------------|-----------------------------------|--------|
| ActionID   | Unique ID of the action perfomed. | String |

## Voicenter Response Example

```
1  {
2    "ErrorCode": "200",
3    "Message": "Success",
4    "ActionID": "14d3b31988b247be8ff5818d1cad3d3"
5  }
```

# Mute by a specific call

A Mute request can be sent to mute a specific call.  
To do so the Mute request must contain the call unique ID – ivrid.

The call ID can be found in real time by using:

### 1. Real-time API

Real-time API provides real-time events of calls and call status changes as well as user status changes in your organization.  
One of the real-time events parameters is “ivrid”.

### 2. Active call API

Active calls API can be used to get the details of every active call upon request.  
One of the active call parameters is “ivrid”.

## URI

```
1 | https://monitor1.voicenter.co/api/MuteUnmuteCalls?ivrid=XXXXXXXXXXXXXXXXXXXX&state=1
```

## Acceptable request types

- 1. POST-JSON
- 2. GET

## Possible Response formats

- 1. POST-JSON

## Client Request Parameters

| Field name | Description            | Type   | Remarks   |
|------------|------------------------|--------|-----------|
| ivrid      | Unique ID of the call. | String | Mandatory |
|            |                        |        |           |

| Field name | Description              | Type   | Remarks   |
|------------|--------------------------|--------|-----------|
| state      | 0 – Unmute.<br>1 - Mute. | String | Mandatory |

Client Request Example (GET)

```
1 | https://YOUR_MONITOR.voicenter.co/api/MuteUnmuteCalls?ivrid=XXXXXXXXXXXXXXXXXXXX&state=1
```

Client Request Example (POST-JSON)

```
1 | {
2 |   "ivrid":"XXXXXXXXXXXXXXXXXXXX",
3 |   "state":"0"
4 | }
```

Voicenter Response Parameters

| Field name | Description  | Type    |
|------------|--|---------|
| ErrorCode  | Describes the status of the client request.<br>The status 200 will indicate a succesfull request.  | Integer |
| Message    | Response status description.<br>Response status description:<br>"Success" – No errors.<br>"UniquelvrID not found" - The Code value or format is invalid.<br>"Parameters are not valid, should be \"ivrid\" and \"mute\" - The sent parameters are invalid. | String  |
| ActionID   | Unique ID of the action perfomed.  | String  |

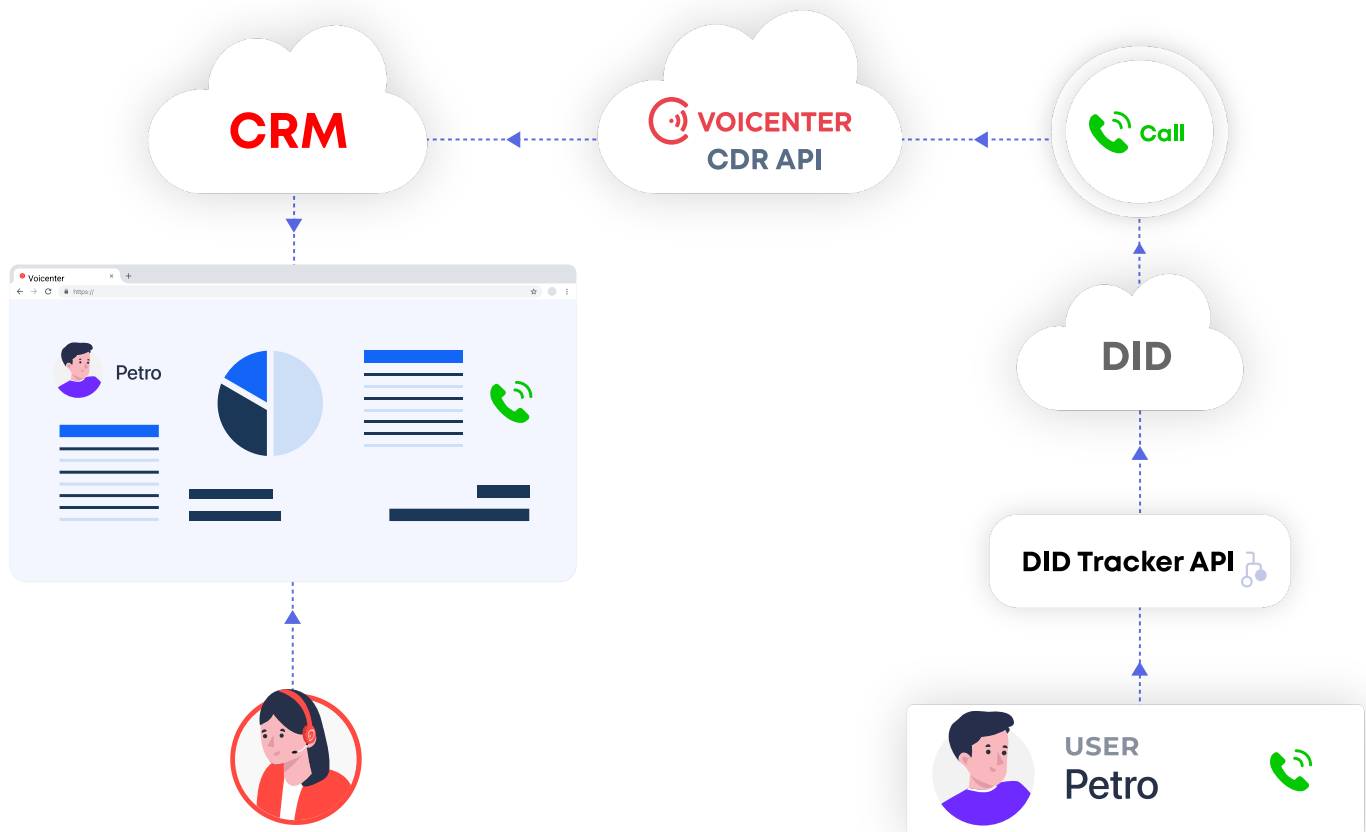
Voicenter Response Example

```
1 | {
2 |   "ErrorCode":"200",
3 |   "Message":"Success",
4 |   "ActionID":"14d3b31988b247be8ff5818d1cad3d3"
5 | }
```



# Lead Tracker API

Lead Tracker API allows Voicenter customers to track incoming calls from your websites, mini-sites, landing pages, etc. It is mainly used when you wish to follow, examine and diagnose how many calls were received from clients who had left their contact info on your various online platforms. Was the caller referred to you from the website via Facebook, Google Ads, etc. Use the following easy-to-implement JS script for advertisement tracking of calling numbers from your websites.



## General Explanation

This API gives you a lot of tools and access to vital information such as which client/lead had left their contact details on your landing page.

Did that client receive a call? And in case they did, did the call help convert the lead to an existing client?

This API can measure the effectiveness of your Google ads, landing pages and any other web advertisement you may have. All you have to do is follow the JS script shown below.

init method accepts the following arguments:

| Argument | Accepts | Example | Description |
|----------|---------|---------|-------------|
|          |         |         |             |

| Argument     | Accepts | Example  | Description  |
|--------------|---------|--|--|
| token        | String  | xxxxxxxxxxxxxxxxxxxxxxxxzzzzzzzzzzzz                           | This Token will be provided by Voicenter and will be assigned to a pool of DIDs (phone numbers) associated under the same “lead tracker” settings. |
| visitorInfo  | object  | {<br>“glid”:"wqer54ytw345uye589y",<br>“fname”:"Chuck Norris" } | Any object which will be send to Lead Tracker API to be temporarily stored and assigned to this DID.   |
| actions      | object  | see examples below   | List of actions to perform on a list of selectors (html elements).   |
| actions.call | array   | see examples below   | List of selectors to which will be applied 'click' listener to make a call (tel: to dial) to a DID.  |
| actions.href | array   | see examples below   | List of selectors which hrefs will be replaced with DID.   |
| actions.text | array   | see examples below   | List of selectors which text will be replaced with DID.  |

## Actions Examples

|   |
|---|
| 1 |
|---|

## Actions Examples

### call\_example.html

You can view this action's CDN here.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta charset="UTF-8">
5  <title>LEAD TRACKER EXAMPLE</title>
6  </head>
7  <body>
8  <button class="call-me">
9  Call Me by class
10 </button>
11 <button id="callMe">
12 Call Me by id
13 </button>
14 <script type="text/javascript" src="https://cdn.voicenter.co/cdn/Scripts/did_trace_worker/index.min.js"></script>
15 <script type="text/javascript">
16 VC_DID_TRACKER.init(
17 'token',
18 {
19 name: 'John Doe'
20 },
21 {
```

```
22   call: ['.call-me', '#callMe']
23   })
24 </script>
25 </body>
26 </html>
```

## href\_example.html

You can view this action's CDN here.

```
1   <!DOCTYPE html>
2   <html lang="en">
3   <head>
4   <meta charset="UTF-8">
5   <title>LEAD TRACKER EXAMPLE</title>
6   </head>
7   <body>
8   <a href="test" class="replace-my-href">
9   I will have a new href by class
10  </a>
11  <br>
12  <a href="test" id="replaceMyHref">
13  I will have a new href by id
14  </a>
15  <script type="text/javascript" src="https://cdn.voicenter.co/cdn/Scripts/did_trace_worker/index.min.js"></script>
16  <script type="text/javascript">
17  VC_DID_TRACKER.init(
18  'token',
19  {
20  name: 'John Doe'
21  },
22  {
23  href: ['.replace-my-href', '#replaceMyHref']
24  })
25  </script>
26  </body>
27  </html>
```

## text\_example.html

You can view this action's CDN here.

```
1   <!DOCTYPE html>
2   <html lang="en">
3   <head>
4   <meta charset="UTF-8">
5   <title>LEAD TRACKER EXAMPLE</title>
6   </head>
7   <body>
8   <p class="replace-me">
9   I will be replaced by class
10  </p>
```

```

11 <p id="replaceMe">
12 I will be replaced by id
13 </p>
14 <script type="text/javascript" src="https://cdn.voicenter.co/cdn/Scripts/did_trace_worker/index.min.js"></script>
15 <script type="text/javascript">
16 VC_DID_TRACKER.init(
17   'token',
18   {
19     name: 'John Doe'
20   },
21   {
22     text: ['.replace-me', '#replaceMe']
23   })
24 </script>
25 </body>
26 </html>

```

## no\_action\_example.html

You can view this action's CDN here.

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <title>LEAD TRACKER EXAMPLE</title>
6 </head>
7 <body>
8 <script type="text/javascript" src="https://cdn.voicenter.co/cdn/Scripts/did_trace_worker/index.min.js"></script>
9 <script type="text/javascript">
10 VC_DID_TRACKER.init(
11   'token',
12   {
13     name: 'John Doe'
14   })
15   .then(function (DID) {
16     console.log(`DID: ${DID}`)
17   })
18 </script>
19 </body>
20 </html>

```

```

1 <script type="text/javascript"
2   src="https://cdn.voicenter.co/cdn/Scripts/did_trace_worker/index.min.js"></script>
3 <script type="text/javascript">

```

or save it on your server and refer to its local location:

## Index.min.js backend file

This javascript file should be included in your HTML file for the function to work.

The content is also available [here](#).

So you can easily copy-paste this in the html body (as shown in previous examples):

```
1  const VC_DID_TRACKER=function(){const t={href:[],text:[],call:[]},e="voicenter_did_tracer_did",n="voicenter_did_tracer_did";return function(){t.call.push({url:e,text:n,call:t.call})}};
```