



# API Documentation

**VOICENTER** DEV

Revision 3.2

# About Voicenter APIs for Developers

Voicenter is a cloud based VoIP telephony solution that offers APIs to integrate your telephony system with your organization's other information systems.

With Voicenter's APIs you can use data and advanced tools to improve your ROI, increase your representatives' efficiency, and provide better service to your clients.

We are certain that our APIs will provide clear advantages for your call center.

1. Click 2 call .....	1-6
2. Pop-up screen .....	7-9
3. Call log .....	10-15
4. CDR notification system .....	16-25
5. External IVR .....	26-41
6. Dialer API .....	42-51
7. Active calls .....	52-54
8. Get organization extension list .....	55-56
9. RealTime .....	57-58
10. Login\ Logout .....	59-60
11. Web RTC .....	61-63
12. External Layer IVR .....	64-66

# Table of contents



1

Click 2 Call

# 1. Click 2 Call

Click 2 Call 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 click 2 call 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.

## 1.1. General Click to Call API description:

### API URL:

<https://api.voicenter.co.il/forwardDialer/click2call.aspx>

### Example:

<https://api.voicenter.co.il/ForwardDialer/click2call.aspx?phone=057XXXXXXX&target=03XXXXXXX&code=XXXXXXXXXXXX&action=call>

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

1. Simple call
2. Dial & play record
3. Conference call

### Acceptable request types:

1. POST
2. GET

### Possible Response formats:

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

1. XML (default)
2. JSON

## 1.2. Call parameters

Parameter	Description	Appearance
<b>Phone</b>	First destination to dial. Phone number or Starkey/ Voicenter extension. Phone should be in E164 format, but without the leading plus (+) sign. If the phone is Israeli, format can be 0***** too. In case of dialing to an extension behind a SIP Trunk, please set Phone=SIPTRUNK. In case of extension dependency on user login please et phone=logincode phonelogincode=[login code]	Mandatory
<b>target</b>	Second destination to dial. Phone number or Starkey/ Voicenter extension. Phone should be in E164 format, but without the leading plus (+) sign. If the Target is Israeli, format can be 0***** too.	Mandatory
<b>code</b>	Caller identity	Mandatory
<b>action</b>	Call (action=call)	Mandatory
<b>record</b>	Record the call Acceptable values: true / false If not provided set to false.	Optional
<b>phonecallerid</b>	Caller id of the phone Not supported by all providers	Optional
<b>phonecallername</b>	Caller name of the phone Not supported by all providers	Optional
<b>targetcallerid</b>	Caller id of the target Not supported by all providers	Optional
<b>targetcallername</b>	Caller name of the target Not supported by all providers	Optional
<b>phonemaxdialtime</b>	phone maximum dial time, in seconds. If not provided set to 60 seconds.	Optional
<b>targetmaxdialtime</b>	target maximum dial time, in seconds. If not provided set to 60 seconds.	Optional
<b>maxduration</b>	Maximum call duration in seconds. If not provided set to 7200 seconds.	Optional
<b>phoneautoanswer</b>	Only for Starkey / Voicenter extensions Expecting for true or 1 for phone automatic answer If not specified set to false	Optional
<b>targetautoanswer</b>	Only for Starkey / Voicenter extensions Expecting for true or 1 for target automatic answer If not specified set to false	Optional
<b>checkphonedevicestate</b>	Block call if extension offline	Optional
<b>checktargetdevicestate</b>	Block call if extension offline	Optional
<b>language</b>	Specify language for system recordings/prompts/etc.. Acceptable format is [he/en/ru/ etc...]	Optional
<b>vm_id</b>	Voicemail code. Only n case phone is Starkey /Voicenter extension	Optional
<b>format</b>	Control the format of the response Possible formats : • XML • JSON The default is XML	Optional
<b>var_*</b>	Any custom parameter with "var_" prefix. For example: var_customer=2342 This parameter used in: • Voicenter chrome notification extension • Voicenter desktop notification program • CDR to CRM API The limit is 10 parameters.	Optional

### 1.3. Message (Dial & Play voice message)

Parameter	Description	Appearance
<b>Phone</b>	First destination to dial Phone number or Starkey / Voicenter extension Phone format can be 0***** as well as 972*****	Mandatory
<b>code</b>	Caller identity	Mandatory
<b>action</b>	Message (action=message)	Mandatory
<b>pre_rec</b>	Name of the record without extension First record to play	Optional
<b>Dtmf</b>	Numeric sequence Will be played number by number	Optional
<b>loop_rec</b>	After -number record ( without extension )	Optional
<b>post_rec</b>	Last record to be played (without extension)	Optional
<b>Record</b>	Record the call. Acceptable variables true / false	Optional
<b>targetcallerid</b>	Caller id of the target Not supported by all providers	Optional
<b>targetcallername</b>	Caller name of the target Not supported by all providers	Optional
<b>phonemaxdialtime</b>	In seconds Phone maximum dial time If not provided set to 30	Optional
<b>maxduration</b>	In seconds, Maximum call duration If not provided set to 30.	Optional
<b>phoneautoanswer</b>	Only for Starkey / Voicenter extensions Expecting for true or 1 for phone automatic answer If not specified set to false	Optional

### 1.4. Callference call

Parameter	Description	Appearance
<b>Phone</b>	First destination to dial Phone number or Starkey / Voicenter extension Phone format can be 0***** as well as 972*****	Mandatory
<b>code</b>	Caller identity	Mandatory
<b>action</b>	Callference (action=callference)	Mandatory
<b>bbbcode</b>	Callference entry code	Optional
<b>targetcallerid</b>	Caller id of the target Not supported by all providers	Optional
<b>targetcallername</b>	Caller name of the target Not supported by all providers	Optional
<b>phonemaxdialtime</b>	In seconds Phone maximum dial time If not provided set to 30.	Optional
<b>maxduration</b>	In seconds Maximum call duration If not provided set to 30.	Optional
<b>phoneautoanswer</b>	Only for Starkey / Voicenter extensions. Expecting for true or 1 for phone automatic answer If not specified set to false	Optional

## 1.5. Terminate

Parameter	Description	Appearance
<b>Phone</b>	Voicenter extension To hangup	Mandatory
<b>code</b>	Caller identity	Mandatory
<b>action</b>	terminate (action=terminate)	Mandatory

## 1.6. Response

Response is XML (XML-RPC structure) or JSON

Fields:

### 1. ERRORCODE (Integer) List

Code	Description
0	OK
1	Invalid request parameters
2	Application error
3	Extension representative offline
4	Extension blocked for Click2Call calls

### 2. ERRORMESSAGE (String)

A short sentence describing the

### 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>

```

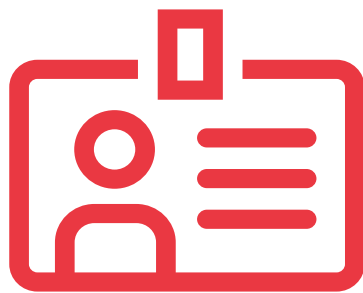


### JSON response Example:

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

### 1.7. PHP Sample ( with XML response )

```
1 $params="code=$code&phone=$voicenter_code_phone&phonecallerid=0500000000&target=$target
2 &targetcallerid=$target&record=true";
3 $ch = curl_init("https://api.voicenter.co.il/ForwardDialer/click2call.aspx");
4 curl_setopt($ch, CURLOPT_POST,1);
5 curl_setopt($ch, CURLOPT_POSTFIELDS , $params);
6 curl_setopt($ch, CURLOPT_RETURNTRANSFER ,1);
7 $response = curl_exec($ch);
8 $callid = return_callid_from_string($response );
9 function return_callid_from_string($str) {
10     $xml_str = simplexml_load_string($str);
11     foreach($xml_str->params->param->value->struct->member as $a){
12         if($a->name == "CALLID") { return $a->value->string; }
13     }
14 }
```



2

POP-UP Screen

## 2. POP UP Screen

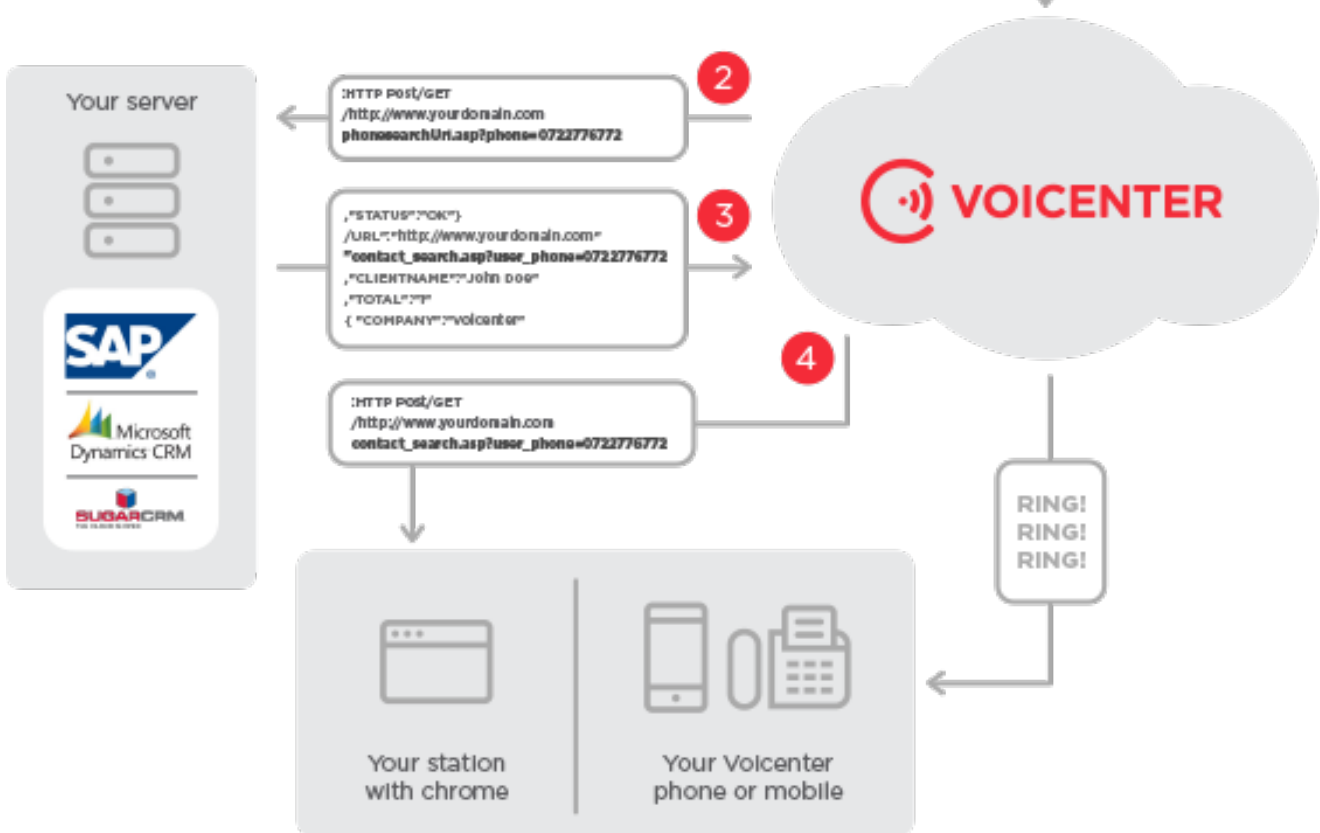
**Voicenter Business cloud communication services proud to present its extension for Chrome give you a simple and powerful Click to call tool that works immediately with your voicenter account allowing you to Right-click any number in the browser and select “Click2Call to #phone number#” in the dropdown menu.**

The Click to call feature can be configured to call any of your Voicenter extensions, your Mobile, Home number – or any other destination you like – connection it to the selected number you wish to dial. Your system will send our Cloud the 2 “Legs” you want it to connect – and the cloud will call them both and connect them together.

In Addition, the extension lets you Pop up a screen with the caller’s contact details right out of your business information software (CRM) – with a simple integration tools for the developers.

Warning explained: When installing the extension it says “This extension will have access to all browsing history and private data on all websites”. This is the default warning for any extension that needs content script access to all websites. This access is needed for detecting phone numbers to call using a click and not typing it yourself. We are obligated to your privacy and are not collecting any personal data or browsing history from your browser.

## How Does Voicenter Pop up screen Work?



- **Step 1:** A caller is calling your phone number. Caller ID is gathered by Voicenter Cloud.
- **Step 2:** Voicenter Cloud sends the caller ID to your Server, which checks if the phone number belongs to one of your clients. For example: <http://www.yourdomain.com/phoneSearchUrl.asp?phone=0722776772>
- **Step 3:** If a contact is found, a JSON is sent back to Voicenter Cloud with the following parameters:

Field name	Description	Type	Remarks
STATUS		String	
URL		String	
CLIENTNAME		String	
TOTAL		String	
COMPANY		String	

```

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

```

- **Step 4:** Voicenter sends to the user's Chrome browser all the details of the call. A bubble with a notification on the call should pop. Pressing the bubble will load a new web browser page with the provided link from Step 3.
- **Step 5:** Voicenter cloud rings voicenter destination extension immediately.



3

Call log

# 3. Call log (Version 0.1)

**CDR API allows you to get call record details by simple WEB request.**

This API allows you to run a specific query for call record details made by a specific user. In order to receive call record details automatically on an ongoing basis, please use the CDR notification API.

**URI:** <https://api1.voicenter.co.il/hub/cdr/>

**Request:**

accepts JSON and GET

**CDR request parameters:**

Parameter	Description
code	String Personal identifier, provided by Voicenter
format	String Format of the response Current possible options: • JSON • HTML
search	Array of search criteria * Search parameters described in following table
fields	Array of strings Describes which fields will be returned in response. Current possible options: • date • fromdate (e.g. 2014-11-18T16:31) • todate • phones • Type • DID • CallerNumber • CallerExtension • TargetNumber • TargetExtension • TargetPrefixName • Duration • Price • RecordURL • RecordExpect • RepresentativeName • RepresentativeCode • DTMFData • CallID • QueueName • DialStatus • DialStatus2 • RingTime • CustomData • DepartmentName • DepartmentId

**Search criteria options are**

Parameter	Description
fromdate	Date time in ISO 8601 format
todate	Date time in ISO 8601 format
phones	Array of strings. Each string is phone number to search include country code
extensions	Array of strings. Each string is Voicenter SIP trunk
IdentityCriteria	One of the following: Account, Hierarchical, Department, User

**Request example in JSON format:**

```

1  {
2  "code": "xxxxxxxxxxxxxxxx",
3  "format": "JSON",
4  "fields": [
5      "Date",
6      "Type",
7      "DID",
8      "CallerNumber",
9      "CallerExtension",
10     "TargetNumber",
11     "Duration",
12     "RecordURL",
13     "RepresentativeName"
14 ],
15 "search": {
16     "fromdate": "2015-05-14T08:00",
17     "todate": "2015-05-14T16:30",
18     "phones": ["97272776772"]
19 }
20 }
```

**Request example in GET format:**

<https://api1.voicenter.co.il/hub/cdr/?code=xxxxxxxxxxxxxxxxxxxxxxxx&format=JSON&todate=2014-11-18T16:31&fromdate=2014-11-11T16:31&phones=972547588535&phones=0500000000&extension=SipSip&fields=Date&fields=Type&Fields=DID&Fields=CallerNumber>

**Response:**

**JSON:**

Parameter	Description
ERROR_NUMBER	Integer Represents error number
ERROR_DESCRIPTION	String Represents error description
CDR_LIST	Array of objects Each object is call detailed record, as requested

**HTML:**

In case of HTML the response is table-based HTML page

**Errors:**

ERROR_NUMBER	ERROR_DESCRIPTION
0	OK
1	Request limit exceeded. Please try again later.
2	Authorization failed.
3	CDR limit exceeded.
4	IP address xx.xx.xx.xx is not trusted.

## CDR request parameters:

#	Field	Description	Example
1	caller	Displays the caller's phone number. The number that will appear at the destination.	"caller": "0722776772"
2	target	Displays the destination of the call. Can be a phone number or the extension SIP code. The phone number value will be sent with the international country prefix.	"target": "AAPINFzL" / "target": "972722776772"
3	time	Displays the time that that the call was made in Epoch time.	"time": 1536855354
4	duration	Displays the duration of the call(seconds). This duration do not include the ringing duration only the actual time of the conversation that was made.	"duration": 33
5	ivruniqueid	Displays the ID code of the specific call.	"ivruniqueid": "201809131615530APIAPI-APIAc40c3d53"
6	type	Displays the type of Call. For example: if it is an incoming/outgoing call? There are several call types.	"type": "Incoming Call" / "type": "Extension Outgoing" / "type": "Click2Callleg1"
7	status	Displays what happened with the specific call? There are several call statuses.	"status": "ANSWER" / "status": "ABANDONE" / "status": "TE"
8	targetextension	Displays the extension SIP code that answered to the incoming call. Sometimes this value will be identical to the "target" field. There are cases that the incoming call is received not directly to the extension, so the "target" can display different value.	"targetextension": "AAPINFzL"
9	callerextension	Displays the extension SIP code that the call was dilled from. This value is different from the "caller" field. In the "caller" field we display the actual number that will appear at the destination.	"callerextension": " AAPINFzL"
10	did	Displays the origin phone number that the caller called to.	"did": "0722776772"
11	queueid	If the call was directed to a queue service, it displays the queue code ID. In case there is no queue, the value will be 0(and not null).	"queueid": 12345
12	queuename	If the call was directed to a queue service, it displays the queue name.	"queuename": "Service Queue"
13	record	Displays a URL link to the call recording.	"record": "http://starkey-centrex-recordings.s3.amazonaws."
14	price	Displays the total price of the call in ILS cents(Agorot).	"price": 7
15	dialtime	Displays the ringing duration of the call(seconds). Not include the actual conversation duration.	"dialtime": 23
16	representative_name	Displays the Voicenter user name that the specific call was associated with.	"representative_name": "Walter Melon"
17	representative_code	Displays the Voicenter user ID code that the specific call was associated with.	"representative_code": " 9996 "
18	targetextension_name	Displays the Voicenter extension name that answered to the specific call.	"targetextension_name": "Walter Melon"
19	callerextension_name	Displays the Voicenter extension name that this specific call was made from.	"callerextension_name": " Walter Melon"
20	target_country	Displays the country name that this outgoing call was made to.	"target_country": "Israel"
21	caller_country	Displays the country name that this incoming call was made from.	"caller_country": "Israel"
22	seconds_waiting_in_queue	This field will only be sent in the json CDR, if the specific called was directed to a queue. It displays the duration(seconds) that the caller waited in the queue.	"seconds_waiting_in_queue": 5
23	OriginalIvrUniqueID	This field will only be sent in the json CDR, if the specific called was related to another call. It displays the origin call code ID. Sometimes calls are transferred in the organization.	"OriginalIvrUniqueID": "201809131730110122APIAPI-API "

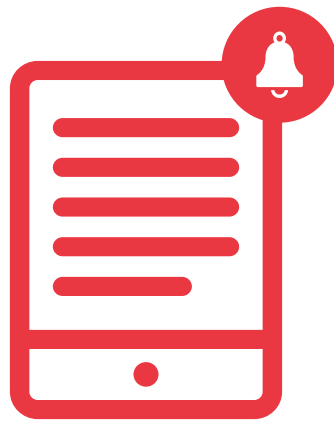


## Call types:

#	CDR type name	Description
1	Incoming Call	A regular outgoing call(manually dialed from the phone).
2	CC	A call that was made through a calling card (Access number) service.
3	Extension Outgoing	A regular outgoing call(manually dialed from the phone).
4	Queue	An incoming call that was received by a queue.
5	Click2Call leg1	A call that was made by click2call(Not by Dialer). Leg 1 of the call. Leg 1 - the initial connection of the call to the extension.
6	Click2Call leg2	A call that was made by click2call(Not by Dialer). Leg 2 of the call. Leg 2 - the actual call that is being made to the destination.
7	VoiceMail	A call that was answered by voicenter voicemail.
8	Callference	A call that was made through Voicenter callference service.
9	XferCDR	A call that manually transferred from an extension to a DID/another extension.
10	ProductiveCall Leg1	A "leg 1" Agents Auto Dialer calls. Leg1 - the initial connection of the call to the extension.
11	ProductiveCall Leg2	A "leg 2" Agents Auto Dialer calls. Leg 2 -the actual call that is being made to the destination.
12	Scrubber	A call that was made through Voicenter's Scrubber service. *Not released yet.
13	Click 2 IVR	"Leg1" Predictive Dialer calls. Leg1 - the initial connection of the call to the extension.
14	Click 2 IVR Incoming	"Leg 2" Predictive Dialer calls. Leg2 - the actual call that is being made to the destination.
15	Click 2 Queue Incoming	"Leg2" Predictive Dialer calls that were made through a queue. Leg2 - the actual call that is being made to the destination.
16	FaxCdr	A call that was made through Voicenter's internal outgoing fax service. *Not released yet.
17	Attended CDR leg1	A call that was transferred with consultation.
18	Attended CDR leg2	A call that was transferred with consultation. 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". The "Leg 2" is the part of the call between "B" and the initial caller.
19	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).

## Call statuses

#	CDR Status Name	Description
1	NOTDIALED	Hang-up occurred before the call was made.
2	ANSWER	A call is answered. A successful dial. The caller reached the callee. Whenever we receive an answer response signal, also when the call reached local voicemail service and etc.
3	BUSY	Busy signal. The dial command reached its number but the number is busy.
4	NOANSWER	No answer. The dial command reached its number, the number rang for too long, then the dial timed out
5	CANCEL	A call is canceled. The dial command reached its number but the caller hung up before the callee picked up.
6	ABANDONE	When using Voicenter's queue service, this status will appear in several cases. A caller hung up before the callee picked up. A caller while waiting in the queue, chose to exit from the queue. A call timeout in the queue.*In the future there will be a call status for each case.
7	VOEND	Hang-up during IVR without actual dialing. In this case, the caller waited in the IVR but hung up before the call rang in any extension.
8	TE	When an incoming call is directed to an IVR recording and afterward it configured to hung up the call.
9	NOTCALLED	A Leg2 Click2Call was not called. When using a click2call service and the caller hung up the call before it dialed at the destination.
10	NOTDIALED	An direct incoming call to an extension that did not reach and dialed at the extension.
11	VOICEMAIL	Call entered to Voicenter voicemail service.
	Error types	This error responses displays cases when there was a problem connecting to the target destinations. It mainly used for Voicenter internal Tracking.
12	CONGESTION	Congestion. This status is usually a sign that the dialed number is not recognized
13	CHANUNAVAIL	Channel unavailable. On SIP, peer may not be registered
14	INVALIDARGS	Error parsing dial command arguments
15	SSWPREAUTH	SSW outgoing call cancel before actual dial



# 4. CDR Notification System

The CDR Notification System is a unique tool from Voicenter that lets you store all of your business telephony information, such as call detail records (CDR), and use/ access it at any time directly from your databases. Our Cloud will send you an HTTP XML-RPC requests that contain detailed information about every call.

## CDR XML RPC specification

This methodology is per the XML-RPC standard, as appearing in the XML-RPC website at <http://www.xml-rpc.org>. Following are some examples representing an XML-RPC post and an XML-RPC response.

### CDR request parameters:

#	Field	Description	Example
1	caller	Displays the caller's phone number. The number that will appear at the destination.	"caller":"0722776772"
2	target	Displays the destination of the call. Can be a phone number or the extension SIP code. The phone number value will be sent with the international country prefix.	"target":"AAPINFzL" / "target": "972722776772"
3	time	Displays the time that that the call was made in Epoch time.	"time":1536855354
4	duration	Displays the duration of the call(seconds). This duration do not include the ringing duration only the actual time of the conversation that was made.	"duration":33
5	ivruniqueid	Displays the ID code of the specific call.	"ivrunique-id":"201809131615530APIAPI-APIAac40c3d53"
6	type	Displays the type of Call. For example: if it is an incoming/outgoing call? There are several call types.	"type":"Incoming Call" / "type":"Extension Outgoing" / "type":" Click2Callleg1"
7	status	Displays what happened with the specific call? There are several call statuses.	"status":"ANSWER" / "status":"ABANDONE" / "status":"TE"
8	targetextension	Displays the extension SIP code that answered to the incoming call. Sometimes this value will be identical to the "target" field. There are cases that the incoming call is received not directly to the extension, so the "target" can display different value.	"targetextension":"AAPINFzL"
9	callerextension	Displays the extension SIP code that the call was dilled from. This value is different from the "caller" field. In the "caller" field we display the actual number that will appear at the destination.	"callerextension": AAPINFzL"
10	did	Displays the origin phone number that the caller called to.	"did":"0722776772"
11	queueid	If the call was directed to a queue service, it displays the queue code ID. In case there is no queue, the value will be 0(and not null).	"queueid":12345
12	queuename	If the call was directed to a queue service, it displays the queue name.	"queuename":"Service Queue"
13	record	Displays a URL link to the call recording.	"record":"http://starkey-centrex-recordings.s3.amazonaws.com/2018091316261401072764f-c3eb6844c-aws-AAPINF-zL-972523258000.mp3"
14	price	Displays the total price of the call in ILS cents(Agorot).	"price":7
15	dialtime	Displays the ringing duration of the call(seconds). Not include the actual conversation duration.	"dialtime":23
16	representative_name	Displays the Voicenter user name that the specific call was associated with.	"representative_name":"Walter Melon"

#	Field	Description	Example
17	representative_code	Displays the Voicenter user ID code that the specific call was associated with.	"representative_code": "9996 "
18	targetextension_name	Displays the Voicenter extension name that answered to the specific call.	"targetextension_name": "Walter Melon"
19	callerextension_name	Displays the Voicenter extension name that this specific call was made from.	"callerextension_name": "Walter Melon"
20	target_country	Displays the country name that this outgoing call was made to.	"target_country": "Israel"
21	caller_country	Displays the country name that this incoming call was made from.	"caller_country": "Israel"
22	seconds_waiting_in_queue	This field will only be sent in the json CDR, if the specific called was directed to a queue. It displays the duration(seconds) that the caller waited in the queue.	"seconds_waiting_in_queue": 5
23	OriginalivrUniqueID	This field will only be sent in the json CDR, if the specific called was related to another call. It displays the origin call code ID. Sometimes calls are transferred in the organization.	"OriginalivrUniqueID": "201809131730110122APIAPI-API "

### Call types:

#	CDR type name	Description
1	Incoming Call	A regular outgoing call(manually dialed from the phone).
2	CC	A call that was made through a calling card (Access number) service.
3	Extension Outgoing	A regular outgoing call(manually dialed from the phone).
4	Queue	An incoming call that was received by a queue.
5	Click2Call leg1	A call that was made by click2call(Not by Dialer). Leg 1 of the call. Leg 1 - the initial connection of the call to the extension.
6	Click2Call leg2	A call that was made by click2call(Not by Dialer). Leg 2 of the call. Leg 2 - the actual call that is being made to the destination.
7	VoiceMail	A call that was answered by voicenter voicemail.
8	Callference	A call that was made through Voicenter callference service.
9	XferCDR	A call that manually transferred from an extension to a DID/another extension.
10	ProductiveCall Leg1	A "leg 1" Agents Auto Dialer calls. Leg1 - the initial connection of the call to the extension.
11	ProductiveCall Leg2	A "leg 2" Agents Auto Dialer calls. Leg 2 -the actual call that is being made to the destination.
12	Scrubber	A call that was made through Voicenter's Scrubber service. *Not released yet.
13	Click 2 IVR	"Leg1" Predictive Dialer calls. Leg1 - the initial connection of the call to the extension.
14	Click 2 IVR Incoming	"Leg 2" Predictive Dialer calls. Leg2 - the actual call that is being made to the destination.
15	Click 2 Queue Incoming	"Leg2" Predictive Dialer calls that were made through a queue. Leg2 - the actual call that is being made to the destination.
16	FaxCdr	A call that was made through Voicenter's internal outgoing fax service. *Not released yet.
17	Attended CDR leg1	A call that was transferred with consultation.
18	Attended CDR leg2	A call that was transferred with consultation. 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". The "Leg 2" is the part of the call between "B" and the initial caller.
19	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).

## Call statuses

#	CDR Status Name	Description
1	NOTDIALED	Hang-up occurred before the call was made.
2	ANSWER	A call is answered. A successful dial. The caller reached the callee. Whenever we receive an answer response signal, also when the call reached local voicemail service and etc.
3	BUSY	Busy signal. The dial command reached its number but the number is busy.
4	NOANSWER	No answer. The dial command reached its number, the number rang for too long, then the dial timed out
5	CANCEL	A call is canceled. The dial command reached its number but the caller hung up before the callee picked up.
6	ABANDONE	When using Voicenter's queue service, this status will appear in several cases. A caller hung up before the callee picked up. A caller while waiting in the queue, chose to exit from the queue. A call timeout in the queue.*In the future there will be a call status for each case.
7	VOEND	Hang-up during IVR without actual dialing. In this case, the caller waited in the IVR but hung up before the call rang in any extension.
8	TE	When an incoming call is directed to an IVR recording and afterward it configured to hung up the call.
9	NOTCALLED	A Leg2 Click2Call was not called. When using a click2call service and the caller hung up the call before it dialed at the destination.
10	NOTDIALED	An direct incoming call to an extension that did not reach and dialed at the extension.
11	VOICEMAIL	Call entered to Voicenter voicemail service.
	Error types	This error responses displays cases when there was a problem connecting to the target destinations. It mainly used for Voicenter internal Tracking.
12	CONGESTION	Congestion. This status is usually a sign that the dialed number is not recognized
13	CHANUNAVAIL	Channel unavailable. On SIP, peer may not be registered
14	INVALIDARGS	Error parsing dial command arguments
15	SSWPREAUTH	SSW outgoing call cancel before actual dial

## CDR response parameters:

#	Name	Type	Example	Description
1	err	int	0	Error code. 0 – OK 1 – Parse error 2 – Application error
2	errdesc	string	OK	Error description

## EXAMPLES

### Xml-rpc:

```
1 <?xml version="1.0"?>
2 <methodCall>
3     <methodName>CDR</methodName>
4     <params>
5         <param>
6             <value>
7                 <struct>
8                     <member>
9                         <name>caller</name>
10                        <value>
11                            <string>0540000000</string>
12                        </value>
13                    </member>
14                    <member>
15                        <name>target</name>
16                        <value>
17                            <string>0500000000</string>
18                        </value>
19                    </member>
20                    <member>
21                        <name>time</name>
22                        <value>
23                            <i8>1345373417</i8>
24                        </value>
25                    </member>
26                    <member>
27                        <name>duration</name>
28                        <value>
29                            <i4>12</i4>
30                        </value>
31                    </member>
32                </struct>
33            </value>
34        </param>
35    </params>
36 </methodCall>
```

```

35         <member>
36             <name>ivruniqueid</name>
37             <value>
38                 <string>sdfsdfsdfsfsf</string>
39             </value>
40         </member>
41         <member>
42             <name>targetextension</name>
43             <value>
44                 <string />
45             </value>
46         </member>
47         <member>
48             <name>callerextension</name>
49             <value>
50                 <string>IuYtReWq</string>
51             </value>
52         </member>
53         <member>
54             <name>did</name>
55             <value>
56                 <string />
57             </value>
58         </member>
59         <member>
60             <name>queueid</name>
61             <value>
62                 <i4>0</i4>
63             </value>
64         </member>
65         <member>
66             <name>queuename</name>
67             <value>
68                 <string />
69             </value>
70         </member>
71         <member>
72             <name>type</name>
73             <value>
74                 <string>Extension Outgoing</
75 string>
76             </value>
77         </member>
78         <member>
79             <name>status</name>
80             <value>
81                 <string>ANSWER</string>
82             </value>
83         </member>
84     </struct>
85 </value>
86 </param>
87 </params>
88 </methodCall>
89
90
91
92
93
94

```



**Valid response:**

```
1 <?xml version="1.0"?>
2 <methodResponse>
3   <params>
4     <param>
5       <value>
6         <struct>
7           <member>
8             <name>err</name>
9             <value>
10              <int>0</int>
11            </value>
12          </member>
13          <member>
14            <name>errdesc</name>
15            <value>
16              <string>OK</string>
17            </value>
18          </member>
19        </struct>
20      </value>
21    </param>
22  </params>
23 </methodResponse>
```

**Error example:**

```
1 <?xml version="1.0" ?>
2 <methodresponse>
3   <params>
4     <param>
5       <value>
6         <struct>
7           <member>
8             <name>err</name>
9             <value>
10              <int>2</int>
11            </value>
12          </member>
13          <member>
14            <name>errdesc</name>
15            <value>
16              <string>Some error description</
17 string>
18            </value>
19          </member>
20        </struct>
21      </value>
22    </param>
23  </params>
24 </methodresponse>
```

**JSON example:**

```
1 {
2   "caller": "0544444444",
3   "target": "0722776772",
4   "time": 1414320455,
5   "duration": 12,
6   "ivruniqueid": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
7   "type": "Extension Outgoing",
8   "status": "ANSWER",
9   "targetextension": "",
10  "callerextension": "SIPSIPSI",
11  "did": "",
12  "queueid": 0,
13  "queueid": 0,
14  "queueid": 0,
15  "queueid": 0,
16  "record": "http://starkey-centrex-recordings.s3.amazonaws.com/xxxxxxxx-
17  aws-0544444444-0722776772.mp3",
18  "dialtime": 31
19 }
```

**Expected response:**

```
1 {
2   "err": "0",
3   "errdesc": "OK"
4 }
```

**C# web service code sample**

CDR Listener implementation in C# asp.net.

We even made an example of a Simple ashx handler attached below

**Simple ashx handler**

Dependencies:

- CookComputing.XmlRpcV2.dll (How-to ...) (c#) (vb.net) )
- Microsoft.NET Framework 4.0

**Code:**

```
1 <%@ WebHandler Language="C#" Class="CDR_Listener" %>
2
3 using System;
4 using System.Web;
5 using CookComputing.XmlRpc;
6
7 //[XmlRpcService(AutoDocumentation=false)]
8 public class CDR_Listener : XmlRpcService
9 {
10     // Change it to true to get more data in output XML.
11     // Change it to false in productive system.
12     private bool isDebug = true;
13
14     private void HandleRequest(CdrData cdr)
15     {
16         /* ADD YOUR CODE HERE */
17     }
18 }
```

```

19
20 [XmlRpcMethod()]
21 public XmlRpcStruct CDR(XmlRpcStruct data)
22 {
23     XmlRpcStruct ret = new XmlRpcStruct();
24     if(this.isDebug)
25         ret.Add("origrequest", data);
26     try
27     {
28         CdrData cdr = new CdrData()
29         {
30             Callerid = GetValue<string>("caller", string.Empty, data),
31             Target = GetValue<string>("target", string.Empty, data),
32             Duration = GetValue<int>("duration", 0, data),
33             IvruniqueID = GetValue<string>("ivruniqueid", string.Empty,
34 data),
35             Epoch = GetValue<long>("time", 0, data),
36             extensionCaller = GetValue<string>("callerextension", string.
37 Empty, data),
38             extensionTarget = GetValue<string>("targetextension", string.
39 Empty, data),
40             status = GetValue<string>("status", string.Empty, data),
41             did = GetValue<string>("did", string.Empty, data),
42             type = GetValue<string>("type", string.Empty, data),
43             queueID = GetValue<int>("queueid", 0, data),
44             queueName = GetValue<string>("queuename", string.Empty, data)
45         };
46         if (this.isDebug)
47             ret.Add("got", cdr);
48
49         try
50         {
51             HandleRequest(cdr);
52         }
53         catch (Exception ex)
54         {
55             if (this.isDebug)
56                 throw new CustomerException(ex, "Customer code
57 exception");
58             else
59                 throw new CustomerException(new Exception("Oops...
60 something wrong"), "Customer code exception");
61         }
62     }
63     catch (CustomerException ex)
64     {
65         ret.Add("err", "2");
66         ret.Add("errdesc", ex.OriginalException.ToString() + " >>> " +
67 ex.Description);
68         return ret;
69     }
70     catch (Exception ex)
71     {
72         ret.Add("err", "1");
73         ret.Add("errdesc", ex.ToString());
74         return ret;
75     }
76
77     ret.Add("err", "0");
78     ret.Add("errdesc", "OK");
79
80     return ret;

```

```
81     }
82
83
84     private T GetValue<T>(string key, T defaultValue, XmlRpcStruct data)
85     {
86         key = key.Trim();
87         if (!data.ContainsKey(key))
88             return defaultValue;
89         T value;
90         try
91         {
92             value = (T)Convert.ChangeType(data[key], typeof(T));
93         }
94         catch
95         {
96             value = defaultValue;
97         }
98         return value;
99     }
100 }
101
102
103
104
105
106 public class CdrData
107 {
108     public string Target = string.Empty;
109     public string Callerid = string.Empty;
110     public string IvrUniqueID = string.Empty;
111     public long Epoch = 0;
112     public int Duration = 0;
113     public DateTime Time { get { return new DateTime(1970, 1, 1, 0, 0, 0,
114 DateTimeKind.Unspecified).AddSeconds(this.Epoch); } }
115     public string extensionTarget;
116     public string extensionCaller;
117     public string did;
118     public string callType;
119     public string callStatus;
120     public int queueID;
121     public string queueName;
122 }
123
124
125
126 public class CustomerException : Exception
127 {
128     public Exception OriginalException { get; private set; }
129     public string Description { get; private set; }
130
131     public CustomerException(Exception OriginalException, string Description)
132     {
133         this.OriginalException = OriginalException;
134         this.Description = Description.Trim();
135     }
136 }
```

### To receive additional value: (REQUIRES IMPLEMENTATION!!!)

1. Add field to CdrData class.
2. "CdrData cdr = new CdrData() { ... // Add parsing row HERE }".
3. Check that the XML includes the new value to be able to get it.



6

External IVR

# 5. External IVR

## 5.1. Method name: GET\_IVR\_ACTION

This method's request is sent by voicenter server to a designated URL on the client's server, with the incoming call details:

### IVR\_MENU REQUEST:

Field name	Description	Type	Remarks
CALLER_ID	Caller ID	String	
CALLER_NAME	Caller name	String	
DID	The number called	String	
MENU	Client's menu identifier	String	will be "-1" at the beginning of the call
PREVIOUS_MENU	Client's menu identifier	String	will be "-1" at the beginning of the call
DTMF	The DTMF the caller pressed	String	
IVR_UNIQUE_ID	A unique ID for the incoming call	String	
IVR_UNIQUE_ID	A unique ID for the incoming call	String	32 chars long

## EXAMPLES

### Xml-rpc:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <methodCall>
3      <methodName>GET_IVR_ACTION</methodName>
4      <params>
5          <param>
6              <value>
7                  <struct>
8                      <member>
9                          <name>DID</name>
10                         <value>
11                             <string>0730000000</string>
12                         </value>
13                     </member>
14                     <member>
15                         <name>CALLER_ID</name>
16                         <value>
17                             <string>0540000000</string>
18                         </value>
19                     </member>
20                     <member>
21                         <name>CALLER_NAME</name>
22                         <value>
23                             <string>017W</string>
24                         </value>
25                     </member>
26                     <member>
27                         <name>IVR_UNIQUE_ID</name>
28                         <value>
29                             <string>ced4619f30fc17182d8e41ddf4
30                             d6869X</string>
31                         </value>
32                     </member>
33                 </struct>

```

```

34         <name>DTMF</name>
35         <value>
36             <string/>
37         </value>
38     </member>
39     <member>
40         <name>MENU</name>
41         <value>
42             <string>-1</string>
43         </value>
44     </member>
45     <member>
46         <name>PREVIOUS_MENU</name>
47         <value>
48             <string>-1</string>
49         </value>
50     </member>
51 </struct>
52 </value>
53 </param>
54 </params>
55 </methodCall>

```

#### JSON:

```

1  {
2      METHOD: "GET_IVR_ACTION",
3      DATA: {
4          DID: "0732793256",
5          CALLER_ID: "n5zx1C2a",
6          CALLER_NAME: "n5zx1C2a",
7          IVR_UNIQUE_ID: "1bcd7954224861f85a2d70612f248282",
8          DTMF: "",
9          MENU: "-1",
10         PREVIOUS_MENU: "-1"
11     }
12 }

```

#### IVR\_MENU REQUEST:

Field name	Description	Type	Appearance
STATUS	0 - OK 1- Error	Integer	Mandatory
ACTION	One of the following options in the next table: • MENU • CALL • HANGUP • SAYDIGITS • VO • SILENTRECORD	String	Mandatory
LANGUAGE	• he • en	String	Mandatory
DATA	Structure contains relevant data of: MENU / CALL / HANGUP /...	Structure	Mandatory

## Action options

MENU - another IVR menu			
Field name	Description	Type	Apearance
REC	The file name of the record to play	String	Mandatory
LOOP	The number of times to loop-play the record	Integer	Mandatory
DELAY	The number of seconds to wait between each loop	Integer	Mandatory
SIZE	The Maximum number of digits from the caller	Integer	Mandatory
MENU	Client's menu identifier	String	Mandatory
VALIDATION	DTMF input validation 1 - validation 0 - no validation	Integer	Mandatory
CALL - Dial and connect any destination world wide			
TARGET	The destination to call	String	Mandatory
CALLERID	The Caller number Shown to the destination	String	Mandatory
CALLERNAME	The Caller name Shown to the destination	String	Mandatory
CALLER_REC	The record the caller will hear before dialing destination	String	Optional
MOH_CLASS	The music class heard while dialing a destination	String	Optional
TARGET_REC	The record the target will hear before connecting to the caller	String	Optional
TARGET_REC	The record the target will hear before connecting to the caller	String	Optional
CALL_MAX_DIAL	Maximum time in seconds to wait while dialing	Integer	Mandatory
CALL_MAX	Maximum duration in seconds for the call	Integer	Mandatory
NEXT_MENU	Next menu the call will be redirected to in case of call failure	String	Optional. If Empty - call will be disconnected.
NEXT_MENU_RECORD	The name of the record of the menu the call will redirected to in case of call failure	String	Optional
NEXT_MENU_LOOP	The number of times to loop-play the record of the next menu	Integer	Optional
NEXT_MENU_SIZE	The Maximum number of digits from the caller	Integer	Optional
NEXT_MENU_WAIT	The number of seconds to wait between each loop, in the next menu	Integer	Optional
REC_STATUS	0 - Don't record 1 - Record call	Integer	Mandatory
DTMF_PLAY	After the called party answers, send digits as a DTMF stream	String	Optional
DIALSTRINGS	Array of ready dialstrings for dial	Array	Optional
SAYDIGITS - Say gigits			
REC	The file name of the record to play	String	Mandatory
LOOP	The number of times to loop-play the record	Integer	Mandatory
DELAY	The number of seconds to wait between each loop	Integer	Mandatory
SIZE	The Maximum number of digits from the caller	Integer	Mandatory
MENU	Client's menu identifier	String	Mandatory
VALIDATION	DTMF input validation 1 - validation 0 - no validation	Integer	Mandatory
SAYDIGITSRECORD	Record to play before DIGITS	String	Mandatory
DIGITS	DIGITS to play • Pay attention to string type! Do not use integer type to prevent zero loose in zero - leading numbers like '0547.....' Any characters other than numbers will be ignored.		Mandatory



SILENTRECORD - Answer the call & record it			
Field name	Description	Type	Apearance
REC	The file name of the record to play before recording starts	String	Optioinal
MAXDURATION	Maximum call duration	Integer	Optioinal
VO - go to IVR layer			
LAYER	Layer number to go to	Integer	Mandatory
RECORD	Record to play	String	Mandatory
LOOP	The number of times to loop-play the record	Integer	Mandatory
DELAY	The number of seconds to wait between each loop	Integer	Mandatory
HANGUP - Hang up the call			
HANGUP_REC	The "good bye" message before hangup.	String	Optional
HANGUP_CAUSE	Cause of the hangup	Integer	Optional

## HANGUP CALL EXAMPLE

### Xml-rpc:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <methodResponse>
3    <params>
4      <param>
5        <value>
6          <struct>
7            <member>
8              <name>STATUS</name>
9              <value> <int>0</int> </value>
10           </member>
11          <member>
12            <name>LANGUAGE</name>
13            <value><string>he</string></value>
14          </member>
15          <member>
16            <name>ACTION</name>
17            <value>
18              <string>HANGUP</string>
19            </value>
20          </member>
21          <member>
22            <name>DATA</name>
23            <value>
24              <struct>
25                <member>
26                  <name>HANGUP_REC</name>
27                  <value>
28                    <string>blank</string>
29                  </value>
30                </member>
31                <member>
32                  <name>HANGUP_CAUSE</name>
33                  <value>

```

```

34         <int>16</int>
35         </value>
36     </member>
37 </struct>
38 </value>
39 </member>
40 </struct>
41 </value>
42 </param>
43 </params>
44 </methodResponse>

```

## JSON:

```

1  {
2    "STATUS" : 0,
3    "ACTION" : "HANGUP",
4    "LANGUAGE" : "HE",
5    "DATA" : {
6      "HANGUP_REC": "blank",
7      "HANGUP_CAUSE": 16
8    }
9  }

```

## CALL EXAMPLE

### Xml-rpc:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <methodResponse>
3    <params>
4      <param>
5        <value>
6          <struct>
7            <member>
8              <name>STATUS</name>
9              <value>
10               <int>0</int>
11             </value>
12            </member>
13            <member>
14              <name>LANGUAGE</name>
15              <value>
16                <string>he</string>
17              </value>
18            </member>
19
20            <member>
21              <name>ACTION</name>
22              <value>
23                <string>CALL</string>
24              </value>
25            </member>
26            <member>

```

```
27     <name>DATA</name>
28     <value>
29         <struct>
30             <member>
31                 <name>TARGET</name>
32                 <value>
33                     <string>0540123456</string>
34                     </value>
35                 </member>
36             <member>
37                 <name>CALLERID</name>
38                 <value>
39                     <string>Unknown</string>
40                     </value>
41             </member>
42             <member>
43                 <name>CALLERNAME</name>
44                 <value>
45                     <string>?????</string>
46                     </value>
47             </member>
48             <member>
49                 <name>CALLER_REC</name>
50                 <value>
51                     <string>now_calling</string>
52                     </value>
53             </member>
54             <member>
55                 <name>MOH_CLASS</name>
56                 <value>
57                     <string>californication</string>
58                     </value>
59             </member>
60             <member>
61                 <name>TARGET_REC</name>
62                 <value>
63                     <string>blank</string>
64                     </value>
65             </member>
66             <member>
67                 <name>CALL_MAX_DIAL</name>
68                 <value>
69                     <int>35</int>
70                     </value>
71             </member>
72             <member>
73                 <name>CALL_MAX</name>
74                 <value>
75                     <int>600</int>
76                     </value>
77             </member>
78             <member>
79                 <name>NEXT_MENU</name>
80                 <value>
81                     <string>SOME-MENU</string>
82                     </value>
83             </member>
84             <member>
85                 <name>NEXT_MENU_RECORD</name>
86                 <value>
87                     <string>record-filename</string>
```

```

88         </value>
89     </member>
90     <member>
91     <name>NEXT_MENU_LOOP</name>
92     <value>
93         <string>3</string>
94     </value>
95 </member>
96 <member>
97 <name>NEXT_MENU_WAIT</name>
98 <value>
99     <string>1</string>
100 </value>
101 </member>
102 <member>
103 <name>NEXT_MENU_SIZE</name>
104 <value>
105     <string>4</string>
106 </value>
107 </member>
108 <member>
109     <name>REC_STATUS</name>
110 <value>
111     <string>1</string>
112 </value>
113 </member>
114 <member>
115     <name>DTMF_PLAY</name>
116 <value>
117     <string>3
118     </string>
119 </value>
120 </member>
121 <member>
122     <name>DIALSTRINGS</name>
123 <value>
124     <array>
125         <data>
126     <value> SIP/972547000000@provider1</value>
127     <value>SIP/972547000000@provider2</value>
128         </data>
129     </array>
130 </value>
131 </member>
132 </struct>
133 </value>
134 </member>
135 </struct>
136 </value>
137 </param>
138 </params>
139 </methodResponse>

```

**JSON:**

```
1  {
2    "STATUS": 0,
3    "ACTION": "CALL",
4    "LANGUAGE" : "HE",
5    "DATA": {
6      "TARGET": "0540123456",
7      "CALLERID": "0500000000",
8      "CALLERNAME": "",
9      "CALLER_REC": "rec",
10     "MOH_CLASS": "musiconhold",
11     "TARGET_REC": "",
12     "CALL_MAX_DIAL": "25",
13     "CALL_MAX": "3200",
14     "NEXT_MENU": "16",
15     "NEXT_MENU_RECORD": "rec",
16     "NEXT_MENU_LOOP": 1,
17     "NEXT_MENU_SIZE": 2,
18     "NEXT_MENU_WAIT": 1,
19     "REC_STATUS": "0",
20     "DTMF_PLAY": "16",
21     "DIALSTRINGS": [
22       "SIP/972547000000@provider1",
23       "SIP/972547000000@provider2"
24     ]
25   }
26 }
```

**MENU EXAMPLE****Xml-rpc:**

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <methodResponse>
3    <params>
4      <param>
5        <value>
6          <struct>
7            <member>
8              <name>STATUS</name>
9              <value>
10               <int>0</int>
11             </value>
12            </member>
13            <member>
14              <name>ACTION</name>
15              <value>
16                <string>MENU</string>
17              </value>
18            </member>
19            <member>
20              <name>LANGUAGE</name>
21              <value>
22                <string>he</string>
23              </value>
24            </member>
25            <member>
26              <name>DATA</name>
```

```

27         <value>
28             <struct>
29                 <member>
30                     <name>REC</name>
31                     <value>
32                         <string>record_filename</string>
33                         </value>
34                     </member>
35                 <member>
36                     <name>LOOP</name>
37                     <value>
38                         <int>2</int>
39                     </value>
40                 </member>
41                 <member>
42                     <name>DELAY</name>
43                     <value>
44                         <int>1</int>
45                     </value>
46                 </member>
47                 <member>
48                     <name>SIZE</name>
49                     <value>
50                         <int>4</int>
51                     </value>
52                 </member>
53                 <member>
54                     <name>MENU</name>
55                     <value>
56                         <string>menu_name</string>
57                         </value>
58                 </member>
59                 <member>
60                     <name>VALIDATION</name>
61                     <value>
62                         <int>1</int>
63                     </value>
64                 </member>
65             </struct>
66         </value>
67     </member>
68 </struct>
69 </value>
70 </param>
71 </params>
72 </methodResponse>

```

**JSON:**

```

1  {
2      "STATUS" : 0,
3      "ACTION" : "MENU",
4      "LANGUAGE" : "HE",
5      "DATA" : {
6          "REC": "record",
7          "LOOP": 4,
8          "DELAY": 3,
9          "SIZE": 2,
10         "MENU": "menu",
11         "VALIDATION": 0
12     }
13 }

```

## SAYDIGITS EXAMPLE

### Xml-rpc:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <methodResponse>
3   <params>
4     <param>
5       <value>
6         <struct>
7           <member>
8             <name>STATUS</name>
9             <value>
10              <int>0</int>
11            </value>
12          </member>
13          <member>
14            <name>ACTION</name>
15            <value>
16              <string>SAYDIGITS</string>
17            </value>
18          </member>
19          <member>
20            <name>LANGUAGE</name>
21            <value>
22              <string>he</string>
23            </value>
24          </member>
25          <member>
26            <name>DATA</name>
27            <value>
28              <struct>
29                <member>
30                  <name>REC</name>
31                  <value>
32                    <string>record_filename</string>
33                  </value>
34                </member>
35                <member>
36                  <name>LOOP</name>
37                  <value>
38                    <int>2</int>
39                  </value>
40                </member>
41                <member>
42                  <name>DELAY</name>
43                  <value>
44                    <int>1</int>
45                  </value>
46                </member>
47                <member>
48                  <name>SIZE</name>
49                  <value>
50                    <int>4</int>
51                  </value>
52                </member>
53                <member>
54                  <name>MENU</name>
55                  <value>
56                    <string>menu_name</string>
57                  </value>
58                </member>
59              </struct>

```

```
60         <name>VALIDATION</name>
61         <value>
62             <int>1</int>
63         </value>
64     </member>
65 </member>
66
67     <name>SAYDIGITSRECORD</name>
68     <value>
69     <string>blank</string>
70     </value>
71 </member>
72 </member>
73     <name>DIGITS</name>
74     <value>
75
76     <string>154654</string>
77     </value>
78 </member>
79 </struct>
80 </value>
81 </member>
82 </struct>
83 </value>
84 </param>
85 </params>
86 </methodResponse>
```

#### JSON:

```
1  {
2  "STATUS" : 0,
3  "ACTION" : " SAYDIGITS ",
4  "LANGUAGE" : "HE",
5  "DATA" : {
6      "REC": "record",
7      "LOOP": 4,
8      "DELAY": 3,
9      "SIZE": 2,
10     "MENU": "menu",
11     "VALIDATION": 0,
12     "VALIDATION": "SAYDIGITSRECORD",
13     "VALIDATION": "DIGITS"
14 }
```



## 5.2. Method name: HANGUP\_CALL

This method request is sent by voicenter server to report the client server at an end of a call.

### HANGUP\_CALL REQUEST:

Field name	Description	Type	Example
CALLER_ID	Caller ID	String	0721234567
DID	The number called	String	0721231231
IVR_UNIQUE_ID	A unique ID for the incoming call	String	aO6Hsr46344rfiEbjUmnpl0324dfjG3F
TARGET	The destination of the call	String	0501234567
CALL_TIME_EPOCH	Hour and date of the call in Linux EPOCH	Integer	123123123
CALL_START_EPOCH	Hour and date of the call flow start in Linux EPOCH	Integer	123123123
IVR_UNIQUE_ID	A unique ID for the incoming call	String	
CALL_DURATION	Total call duration in seconds	Integer	243
CALL_DAILTIME	Dialing time until destination answered in seconds	Integer	12
CALL_STATUS	Status of the call	String	ANSWERED

### CALL\_STATUS options:

- **ANSWER:** Call is answered. A successful dial. The caller reached the callee.
- **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:** Call is cancelled. The dial command reached its number but the caller hung up before the callee picked up.
- **CONGESTION:** Congestion. This status is usually a sign that the dialled number is not recognised.
- **CHANUNAVAIL:** Channel unavailable. On SIP, peer may not be registered.
- **DONTCALL:** Privacy mode, callee rejected the call
- **TORTURE:** Privacy mode, callee chose to send caller to torture menu
- **INVALIDARGS:** Error parsing Dial command arguments

## SIP RFC error codes

Definition	SIP Equivalent	Cause code (ISUP)
Unallocated (unassigned) number	404 Not Found	1
no route to network	404 Not found	2
no route to destination	404 Not found	3
normal call clearing	BYE or CANCEL (*)	16
user busy	486 Busy here	17
no user responding	408 Request Timeout	18
no answer from the user	480 Temporarily unavailable	19
subscriber absent	480 Temporarily unavailable	20
call rejected	403 Forbidden (+)	21
number changed (w/o diagnostic)	410 Gone	22
number changed (w/ diagnostic)	301 Moved Permanently	22
redirection to new destination	410 Gone	23
non-selected user clearing	404 Not Found (=)	26
destination out of order	502 Bad Gateway	27
address incomplete	484 Address incomplete	28
facility rejected	501 Not implemented	29
normal unspecified	480 Temporarily unavailable	31
no circuit available	503 Service unavailable	34
network out of order	503 Service unavailable	38
temporary failure	503 Service unavailable	41
switching equipment congestion	503 Service unavailable	42
resource unavailable	503 Service unavailable	47
incoming calls barred within CUG	403 Forbidden	55
bearer capability not authorized	403 Forbidden	57
bearer capability not presently	503 Service unavailable	58
bearer capability not implemented	488 Not Acceptable Here	65
only restricted digital avail	488 Not Acceptable Here	70
service or option not implemented	501 Not implemented	79
user not member of CUG	403 Forbidden	87
incompatible destination	503 Service unavailable	88
recovery of timer expiry	504 Gateway timeout	102
protocol error	500 Server internal error	111
interworking unspecified	500 Server internal error	127

## HANGUP\_CALL REQUEST:

Field name	Description	Type	Mandatory
STATUS	0 - OK 1 - Error	Integer	Mandatory

## REQUEST EXAMPLE

### Xml-rpc:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <methodCall>
3   <methodName>HANGUP_CALL</methodName>
4   <params>
5     <param>
6       <value>
7         <struct>
8           <member>
9             <name>DID</name>
10            <value>
11              <string>0730000000</string>
12            </value>
13          </member>
14          <member>
15            <name>CALLER_ID</name>
16            <value>
17              <string>0540000000</string>
18            </value>
19          </member>
20          <member>
21            <name>IVR_UNIQUE_ID</name>
22            <value>
23              <string>ced4619f30fc17182d8e41ddf4d6869X</string>
24            </value>
25          </member>
26          <member>
27            <name>TARGET</name>
28            <value>
29              <string>0500000555</string>
30            </value>
31          </member>
32          <member>
33            <name>CALL_TIME_EPOCH</name>
34            <value>
35              <int>15616841651</int>
36            </value>
37          </member>
38          <member>
39            <name>CALL_START_EPOCH</name>
40            <value>
41              <int>15616841612</int>
42            </value>
43          </member>
44          <member>
45            <name>CALL_DURATION</name>
46            <value>
47              <int>125</int>
48            </value>
49          </member>
50          <member>
51            <name>CALL_DAILTIME</name>
52            <value>
53              <int>15</int>
54            </value>
```

```

55         </member>
56         <member>
57             <name>CALL_STATUS</name>
58             <value>
59                 <string>ANSWERED</string>
60             </value>
61         </member>
62     </struct>
63 </value>
64 </param>
65 </params>
66 </methodCall>

```

#### JSON:

```

1  {
2    "METHOD" : "HANGUP_CALL",
3    "DATA" : {
4        "DID": "0730000000",
5        "CALLER_ID": "0540000000",
6        "IVR_UNIQUE_ID": "ced4619f30fc17182d8e41ddf4d6869X",
7        "TARGET": "0500000555",
8        "CALL_TIME_EPOCH": 15616841651,
9        "CALL_START_EPOCH": 15616841612,
10       "CALL_DURATION": 650,
11       "CALL_DAILTIME": 15,
12       "CALL_STATUS": "ANSWERED"
13     }
14 }

```

## RESPONSE EXAMPLE

#### Xml-rpc:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <methodResponse>
3      <params>
4          <param>
5              <value>
6                  <struct>
7                      <member>
8                          <name>STATUS</name>
9                          <value>
10                             <int>0</int>
11                         </value>
12                     </member>
13                 </struct>
14             </value>
15         </param>
16     </params>
17 </methodCall>

```

#### JSON:

```

1  {
2    "STATUS": 0
3  }

```



6

Dialer API

# 6. Dialer API

## 6.1. Add Call to campaign;

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/AddCall>

### Parameter list:

Name	Type	Appearance	Description
Campaign	string	mandatory	Campaign unique code.
Target	phone	mandatory	Campaign target, usually phone number of the customer / client / etc.
Callerid	phone	optional	<p>Caller identification for the target (phone number that customer will see on his phone display).</p> <p>If not appears or comes empty, the dialer system will try to apply caller id in below priority:</p> <ol style="list-style-type: none"> <li>1. Try to find dynamic caller id for target country.</li> <li>2. Get caller id from worker's extension.</li> <li>3. Get caller id from campaign settings.</li> </ol> <p>In case you want to hide caller id use "Anonymous"</p>
OriginateTime	epoch	optional	Time to originate the call in Unix time epoch. Not available on all campaign types.
CustomerName	string	optional	Name of the customer
Priority	integer	optional	Priority of the call in current campaign. Calls with higher priority will be originated before others.
var_CustomData	string	optional	<p>Any custom data.</p> <p>Will be used in popup notification or CDR to CRM systems. When using GET/POST, send all custom variables with "var_" prefix. In case of JSON usage send all custom variables with "var_" prefix as part of "CustomData" object.</p>

## REQUEST EXAMPLES

### JSON

```

1  {
2    "Campaign": "xxxxxxxx",
3    "Target": "0502454545",
4    "CustomerName" : "John Doe"
5    "Priority" : 42,
6    "CustomData": {
7      "var_test1": 1,
8      "var_test2": "some value"
9    }
10 }
```

### GET

```

1 https://api.voicenter.co.il/ForwardDialer/Dialer/AddCall?Campaign=xxxx&Target=yyyyy&-
2 var_test1=1&var_test2=somevalue
```

### RESPONSE

```

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

ErrorCode	Description
0	OK
1	Invalid campaign code
2	The target is invalid
3	Unexpected internal error
4	Invalid caller id

## 6.2. Remove Call from campaign

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/RemoveCall>

### EXAMPLES

#### Request

```
1 | {
2 |   "Campaign": "xxxxxxxxx",
3 |   "Target": "0502454545",
4 | }
```

#### Response

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

## 6.3. Get campaign list

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/GetCampaignList>

### EXAMPLES

#### Request

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

#### Response

```
1 | {
2 |   "Data": [
3 |     {
4 |       "Name": "Campaign 1",
5 |       "StatusName": "Enabled",
6 |       "Code": "skdfvnjsivbniviv"
7 |     },
8 |     {
9 |       "Name": "Campaign leads",
10 |      "StatusName": "Disabled",
11 |      "Code": "afmksvosvo"
12 |     }
13 |   ],
14 |   "ErrorCode": 0,
15 |   "Description": "OK"
16 | }
```

#### Possible campaign statuses:

1. Enabled
2. Disabled
3. Deleted

## 6.4. Get Members list

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/GetMembersList>

### EXAMPLES

#### Request

```
{
  "Campaign": "adsdasdasd"
}
```

#### Response

```
1  {
2    "Data": [
3      {
4        "Member": "syhrhrhr",
5        "DisplayName": "Support 1",
6      },
7      {
8        "Member": "sssssssss",
9        "DisplayName": "Support 8",
10     }
11   ],
12   "ErrorCode": 0,
13   "Description": "OK"
14 }
```

## 6.5. Remove member

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/RemoveMember>

### EXAMPLES

#### Request

```
1  {
2    "Campaign": "adsdasdasd"
3    "Member": "adsdasdasd"
4  }
5  }
```

#### Response

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



## 6.6. Add member

Accepts GET, POST and JSON

**URL:** <https://api.voicenter.co.il/ForwardDialer/Dialer/AddMember>

### EXAMPLES

#### Request

```
1 | {
2 |     "Campaign": "adsdasd"
3 |     "Member": "adsdasd"
4 | }
```

#### Response

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

## 6.7. Stop campaign

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/StopCampaign>

### EXAMPLES

#### Request

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

#### Response

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

## 6.8. Start Campaign

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/StartCampaign>

### EXAMPLES

#### Request

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

#### Response

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

## 6.9. UpdateCampaignStrategy

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/UpdateCampaignStrategy>

### EXAMPLES

#### Request

```
1 | {  
2 |   "Campaign" : "xxxxxxxxxxxxxxxxxxxxxxxx",  
3 |   "setQueueForMonitoring" : 10000,  
4 |   "setQueueMaximumCallers" : 0,  
5 |   "setCoefficient" : 1  
6 | }
```

#### Response

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

## 6.10. ClearCampaignCalls

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/ClearCampaignCalls>

### EXAMPLES

#### Request

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

#### Response

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

## 6.11. AddCallsBulk

Accepts JSON only !!!

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/AddCallsBulk>

### EXAMPLES

#### Request

```
1 [
2   {
3     "Campaign": "xxxxxxxxxxxxxxxxxxxxxxx",
4     "Target": "0502454545",
5     "CallerID": "072776772",
6     "CallerName": "Dialer",
7     "CustomerName": "Customer 1",
8     "Priority": 1,
9     "CustomData": {
10      "var_test1": 1,
11      "var_tesy2": "some value"
12    }
13  },
14  {
15    "Campaign": "xxxxxxxxxxxxxxxxxxxxxxx",
16    "Target": "0502454545",
17    "CallerID": "072776772",
18    "CallerName": "Dialer",
19    "CustomerName": "Customer 2",
20    "Priority": 2,
21    "CustomData": {
22      "var_test1": 1,
23      "var_tesy2": "some value"
24    }
25  }
26 ]
```

#### Response

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

## 6.12. GetCampaignPendingCalls

Accepts JSON only !!!

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/GetCampaignPendingCalls>

### EXAMPLES

#### Request

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

#### Response

```
1 {
2   "Data": {
3     "Calls": [
4       {
5         "Phone": "972544444444",
6         "CallerID": "",
7         "CallerName": "",
8         "OriginateTime": 0,
9         "Priority": 0,
10        "CustomerName": "Customer",
11        "CallStatus": {
12          "Status": 1,
13          "Description": "Pending"
14        },
15        "CustomData": {
16          "var_test1": 1,
17          "var_tesy2": "some value"
18        }
19      },
20      {
21        "Phone": "972544444444",
22        "CallerID": "",
23        "CallerName": "",
24        "OriginateTime": 0,
25        "Priority": 0,
26        "CustomerName": "Customer 1",
27        "CallStatus": {
28          "Status": 1,
29          "Description": "Pending"
30        },
31        "CustomData": {
32          "var_test1": 2,
33          "var_tesy2": "some value 2"
34        }
35      }
36    ],
37    "Campaign": {
38      "Name": "Test campaign",
39      "StatusName": "Enabled",
40      "TotalPendingCalls": 2,
41      "MaxPriority": 0,
42      "MinPriority": 0,
43      "Code": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
44    }
45  },
46  "ErrorCode": 0,
47  "Description": "OK"
48 }
```

## 6.13. UpdateCall

**Parameter list:** see AddCall

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/UpdateCall>

### EXAMPLES

#### Request

```
1 | {
2 |   "Campaign": "xxxxxxxx",
3 |   "Target": "0502454545",
4 |   "CustomerName" : "John Doe",
5 |   "Priority" : 42,
6 |   "CustomData": {
7 |     "var_test1": 1,
8 |     "var_tesy2": "some value"
9 |   }
10 | }
```

#### GET Request

```
1 | https://api.voicenter.co.il/ForwardDialer/Dialer/UpdateCall?Campaign=xxxx&Tar-
2 | get=yyyy&var_test1=1&var_tesy2=somevalue
```

#### Response

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

## 6.14. GetCampaignDetails

Accepts GET, POST and JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/GetCampaignDetails>

### EXAMPLES

#### JSON Request

```
1 {
2
3   "Campaign" : " xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx "
4
5 }
```

#### JSON Response example

```
1 {
2   "Data": {
3     "Name": "Morning messages",
4     "StatusName": "Enabled",
5     "TotalPendingCalls": 1,
6     "MaxPriority": 7,
7     "MinPriority": 7,
8     "Code": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
9   },
10  "ErrorCode": 0,
11  "Description": "OK"
12 }
```

## 6.15. ConfigureIvrCampaign

Accepts JSON

**URI:** <https://api.voicenter.co.il/ForwardDialer/Dialer/ConfigureIvrCampaign>

### EXAMPLES

#### Request

```
1 {
2   "Campaign": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
3   "ConcurrentStrategy": {
4     "setMaxCalls" : 5
5   },
6   "RepresentativeList": [
7     "SIP1",
8     "SIP2",
9     "SIP3"
10  ]
11 }
```

#### Response

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



7

Active calls

# 7. Active calls

## 7.1. Extensions active calls

**URL:** <https://monitor.voicenter.co.il/comet/API/GetExtensionsCalls?code=xxxxxxxxxx>

### EXAMPLES

```
1  {
2  ERR: 0,
3  DESC: "OK",
4  EXTENSIONS:
5  [
6    {
7      name: "Desk 1",
8      representative: "Dodod",
9      peerstatus: "Reachable",
10     username: "aaaaaaa",
11     calls:
12     [
13       {
14         callername: "0547777777",
15         callerphone: "0547777777",
16         callstarted: 1402233117,
17         callanswered: 1402233123,
18         answered: 1,
19         callstatus: "Talking",
20         ivrid: "xxxxXxxxXxxxXxxxXxxxXxxxXxxxXxxx",
21         direction: "Outgoing"
22       }
23     ]
24   },
25   {
26     name: "Desk 2",
27     representative: "",
28     peerstatus: "Unknown",
29     username: "ttttttt",
30     calls: [ ]
31   }
32 ]
33 }
```

### BAD ANSWERS:

```
1  {
2    "ERR": 1,
3    "DESC": "Authorization failed."
4  }
```

```
1  {
2    "ERR": 2,
3    "DESC": "Internal error."
4  }
```



## 7.2. Queue active callers

**URL:** <https://monitor.voicenter.co.il/comet/API/GetQueuesCallers?code=xxxxxxxxxx>

### EXAMPLES

```
1 {
2   ERR: 0,
3   DESC: "OK",
4   QUEUES:
5   [
6     {
7       Name: "Sales",
8       ID: 10035,
9       Callers:
10      [
11        {
12          Phone: "0570000000",
13          CallID: "xxxxXxxxXxxxXxxxXxxxXxxxXxxxXxxx",
14          JoinTime: 1402235247
15        }
16      ]
17    }
18  ]
19 }
```

### BAD ANSWERS:

```
1 {
2   "ERR": 1,
3   "DESC": "Authorization failed."
4 }
```

```
1 {
2   "ERR": 2,
3   "DESC": "Internal error."
4 }
```



## 8. Get Organizational Extension List

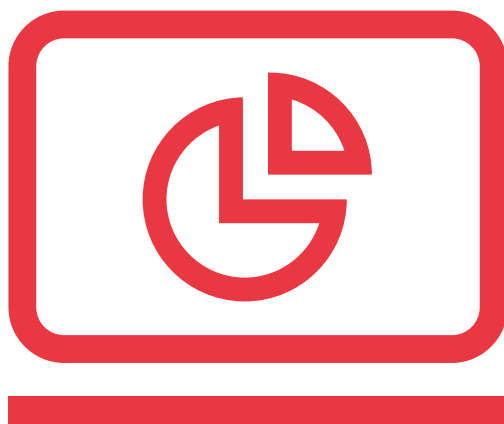
Accepts GET, POST and JSON

**URL:** <https://monitor.voicenter.co.il/Comet/api/GetExtensions?code=XXXXXXXXXXXX&showAll=0>  
code -> user code

showAll -> true for receive complete organization list  
false for current account only

### EXAMPLE

```
1 {
2   ERR: 0,
3   DESC: "OK",
4   EXTENSIONS: [
5     {
6       SIP: "xxxxxxx",
7       Name: "Account name",
8       SpeedDial: "301",
9       AccountID: 1,
10      AccountName: "Acc 1"
11    },
12
13    {
14      SIP: "yyyyyyy",
15      Name: "Account name 2",
16      SpeedDial: "302",
17      AccountID: 2,
18      AccountName: "Acc 2"
19    }
20  ]
21 }
```



9

RealTime

# 9. Real Time

Accepts GET, POST and JSON

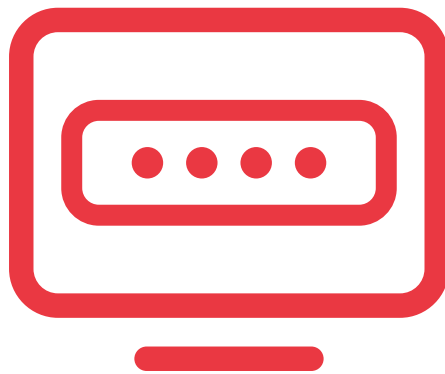
**URL:** <https://monitor.voicenter.co.il/Comet/api/GetExtensions?code=XXXXXXXXXXXX&showAll=0>

code -> user code

showAll -> true for receive complete organization list  
false for current account only

## EXAMPLE

```
1 <html>
2   <head></head>
3   <body>
4     Status: <span id="status-txt">Disconnected</span><br /><br />
5     <div id="response-body"></div>
6   </body>
7 </html>
8 <script src="https://voicenter-voicenterltd.netdna-ssl.com/voicenterMonitorListener.
9 js"></script>
10 <script>
11 window.voicenterAsyncLoad = function() {
12   voicenterListener = new voicenterMonitor();
13   // Connect to server
14   voicenterListener.connect();
15   voicenterListener.onConnect = function(errorCode) {
16     // Success
17     if (errorCode == 0) {
18       document.getElementById("status-txt").innerHTML = "Connecting...";
19       voicenterListener.loginUserCode("XXXXXXXXXXXXXXXXXXXXXXXXXXXX");
20     }
21     // Failed
22     else {
23       document.getElementById("status-txt").innerHTML = "Failed to connect";
24     }
25   };
26   voicenterListener.onLoginResponse = function(errorCode, errorDesc) {
27     // Success
28     if (errorCode == 0) {
29       document.getElementById("status-txt").innerHTML = "Connected";
30       console.log(errorDesc);
31     }
32     // Failed
33     else
34       document.getElementById("status-txt").innerHTML = errorDesc;
35   };
36   // Extension event
37   voicenterListener.onExtensionEvent = function(res) {
38     console.log(res);
39     var cur = document.getElementById("response-body");
40     cur.innerHTML = cur.innerHTML + res.reason + "<br />";
41   };
42   // After logging in, receive all extension info
43   voicenterListener.onAllExtensionsEvent = function(data) {
44     console.log(data);
45   };
46 };</script>
```



10

Login/Logout

# 10. Login/Logout

Accepts GET, POST and JSON

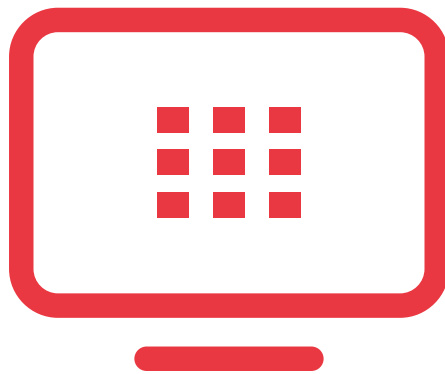
**URL:** <http://betaapi.voicenter.co.il/UserLogin/SetStatusFromAPI>

## Request parameters:

#	Name	Type	Example	Description
1	UserID	string		First 5 digits of User Code in the CPanel. Also shown when hovering a user's name in the CPanel.
2	Status	int	1 - login 2 - logout 3 4 5 7 9 11 12 13 The names of these statuses can be changed in the CPanel.	The user's status in the system.
3	Code	int		User code, for authorization only, can be any code of any user in the organization
4	ExtensionUser	string		The relevant extension sip name to make the action with

## Response parameters:

#	Name	Type	Example	Description
1	Status	int	1 = ok 2 = Authorization failed 3 = Missing parameters 4 = Login error 5 = Initiation Error	
2	StatusError	string	0 - No status error Status 4 errors: 1. Status Changed 2. Failed - Hour Log Error 3. Failed - Queue registration error 4. Extension is already in use 5. You're already logged in with a different extension Status 5 errors: 1. Please choose extension 2. Successfully Logged In 3. Not logged in 4. Wrong Username or Password 5. There was a problem with the user, please contact support 6. An unexpected error occurred 7. Failed - Extension Monitor doesn't exist 8. Failed - User has no organizational PIN Code	In case of Status 4, this is internal error code
3	StatusErrorMessage	string		Description of errors





# 11. WebRTC Web-phone (BETA 1.00)

**Supported browsers:** Firefox, Chrome

**URL:** <http://voicenter.voicenterltd.netdna-cdn.com/voicenterWebPhone-min.js>

## EXAMPLE

### Create voicenterWebPhone object

```
1 | if(!myPhone) {//Important not to recreate a new webphone on an existing webphone object
2 |     myPhone = new voicenterWebPhone();
3 | }
```

### Set callback functions

```
1 | myPhone.onRegister = function(){
2 |     // on extension registered
3 | };
4 | myPhone.onInitiateFailed = function(){
5 |     // failed initiating extension against voicenter servers
6 | };
7 | myPhone.onRegisterFailed = function(){
8 |     // failed registering
9 | };
10 | myPhone.onDisconnect = function(){
11 |     // on extension disconnected
12 | };
13 | myPhone.onRinging = function(ivrID){
14 |     // on call received, ivrID = Unique voicenter call ID
15 | };
16 | myPhone.onCall = function(ivrID){
17 |
18 |     // on initiating call, receive the call ivrID = Unique voicenter call ID
19 |     myPhone.onCallEnd = function(ivrID){
20 |         // call ended
21 |     };
22 |     myPhone.onCallFailed = function(ivrID){
23 |         // failed initiating a call
24 |     };
25 |     myPhone.onCallRejected = function(ivrID){
26 |         // call rejected
27 |     };
28 |     myPhone.onCallAnswer = function (token)
29 |         // on call answer
30 |     };
```

## Methods

```
1 myPhone.initiate({
2     user: "SIPUSERNAME", // Extension user
3     password: "EXTENSIONPASSWORD" // Extension pass
4 });
5
6 // make a call, if success "oncall" event will be called with the relevant ivrID
7 myPhone.call("0500000000");
8
9 // answer a ringing call, function expects the ivrID of the call to answer
10 myPhone.answer(ivrID);
11
12 // hang up a call, function expects the ivrID of the call to hang up
13 myPhone.hangUp(ivrID);
14
15 // mute a call so the other side won't hear you, function expects the ivrID of the
16 // call to mute
17 myPhone.mute(ivrID);
18
19 // unmute a call so the other side will hear you, function expects the ivrID of the
20 // call to unmute
21 myPhone.unmute();
```



# 12. External Layer IVR

## 12.1. Introduction to IVR Layer Web service

This API gives the developer the ability to change IVR Input behavior through a simple API response.

The system will manage a URL and will provide the ability to attach layer input to a URL record. When a caller inserts a DTMF (digit) in an IVR layer which is attached to a URL the system will send an “IvrExternalLayerInput” request with information about the call, and will wait for a response which will specify the result of that specific input.

Additionally, the response could give the system an array of key-value pairs for the system to remember and return in the POPUP and CDR Notification APIs for identification purposes.

### Field Types:

Field name	Description	Type	Remarks
DID	The number that recieved the call	String	
CALLER_ID	Incoming customer caller ID	String	
IVR_UNIQUE_ID	A unique ID for the incoming call	String	32 chars long
DTMF	DTMF the caller pressed	String	
LAYER_ID	Current layer number	Integer	
PREVIOUS_LAYER_ID	Previous layer number	Integer	
STATUS	0 - OK 1 - Error	Integer	Mandatory
ACTION	One of the following options in the next table: • DIAL • GO_TO_LAYER	String	Mandatory
LAYER	Layer number to go to (Mandatory)	Integer	
CALLER_NAME	The Caller name Shown to the destination	String	
CUSTOM_DATA	Array of dynamic list of pairs of key-value (Mandatory)	String	Empty array in case no custom data
NEXT_VO_ID	Next layer the call will be redirected to in case of call failure	Integer	
UNAVAILABLE_VO	Layer number in case of call has no answer	Integer	

## EXAMPLES

### Json:

```
1 Request :
2 {
3     "METHOD": "IVR_LAYER_INPUT",
4     "DATA": {
5         "DID": "0732793256",
6         "CALLER_ID": "0523574321",
7         "IVR_UNIQUE_ID": "xxxxxxxxxxxxxxxxxxxxxxxxxxxx",
8         "DTMF": "1234567",
9         "LAYER_ID": 8,
10        "PREVIOUS_LAYER_ID": 4
11    }
12 }
13
14 Go to layer response:
15 {
16     "STATUS": 0,
17     "ACTION": "GO_TO_LAYER",
18     "LAYER ": 22,
19     "CALLER_NAME": "Shlomi",
20     "CUSTOM_DATA": {
21         "xxx": "111",
22         "yyy ": "222",
23         "zzz": "333"
24     }
25 }
26
27 Dial response:
28 {
29     "STATUS": 0,
30     "ACTION": "DIAL",
31     "MAX_CALL_DURATION": 7200,
32     "MAX_DIAL_DURATION": 30,
33     "RECORDING": "yes",
34     "TARGETS": [
35         TARGET: "0501234567",
36         TYPE: "PHONE"
37     ],
38     "CALLER_ID": "0722776772",
39     "CALLER_NAME": "Voicenter",
40     "NEXT_VO_ID": 1,
41     "ANSWER_NEXT_VO_ID": 1,
42     "UNAVAILABLE_VO_ID": 2,
43     "CUSTOM_DATA": {
44         "xxx": "111",
45         "yyy ": "222",
46         "zzz": "333"
47     }
48 }
49
50
51
52
```