



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

# Web Services API Reference

[StartContactCenterSession](#)

---

# StartContactCenterSession

This operation is part of the [Voice API](#) section of the [Web Services API](#).

## Overview

Initializes a session for the user with the provided parameters. Almost all client applications that provide a full set of voice features should send this operation at login and call [EndContactCenterSession](#) when the application exits.

|                          |             |
|--------------------------|-------------|
| <b>Request URL</b>       | /api/v2/me/ |
| <b>HTTP method</b>       | POST        |
| <b>Required features</b> | api-voice   |

## Parameters

| Parameter     | Value   |
|---------------|---|
| operationName | StartContactCenterSession   |
| channels      | <p>An array of channels the agent uses in the session. Valid channel names are <code>voice</code> and any valid multimedia channel (for example, <code>email</code>, <code>chat</code>, <code>workitem</code>, <code>IM</code> session).</p> <p>If an error occurs on a given channel, Web Services returns a response with the error <code>statusCode</code> and information about what went wrong. For example, if you <code>StartContactCenterSession</code> with <code>voice</code> and <code>chat</code> and your Interaction Server is out of service, Web Services returns a <b>partial success</b> <code>statusCode</code> of 7. The successful part of the operation (<code>voice</code>) returns <code>statusCode:0</code> while the failed part of the operation (<code>chat</code>) returns a <code>statusCode</code> and <code>statusMessage</code> that provide information about why it failed.</p> <p>If all channel operations fail, Web Services only returns a <code>statusCode</code> and <code>statusMessage</code> for one of the channels.</p> |
| place         | <p>This optional parameter specifies the place that should be used for the session.</p> <p>If not specified, then Web Services uses the default place for the agent.</p> <p>If specified and the place contains one device, then Web Services logs in the user on that device for the duration of the session. If the place contains more than one device, then Web Services logs in the user on the <code>ACDPosition</code>. In this scenario, if the place doesn't contain an <code>ACDPosition</code> or if it has multiple</p>   |

| Parameter          | Value  |
|--------------------|--|
|                    | <p>ACDPositions, then the StartContactCenterSession operation fails. <b>Note:</b> If there are multiple ACDPositions, you should specify the devicePath.</p> <p>When the client ends the session, Web Services restores the default device assignment.</p>   |
| loginCode          | This optional parameter specifies the switch login code that should be used for login. If it's not specified, Web Services looks for an appropriate switch login that is assigned to the user for the T-Server.  |
| agentLoginPassword | This optional parameter is passed to RequestAgentLogin if loginCode is specified in the request. This allows the agent to log in to the switch from their desktop rather than having to log in using their physical phone. <b>Note:</b> If the loginCode is not specified in the request, Web Services gets the password from Configuration Server (if it exists) during the appropriate login code calculation and uses it in RequestAgentLogin.  |
| queue              | This optional parameter specifies a queue to be included in the login request.   |
| devicePath         | A path to a device assigned to an agent. This attribute is mandatory when an agent is assigned multiple devices in the same location where the device to be used for login can't be resolved — for example, the place has multiple extensions or multiple ACDPositions. The devicePath specifies which device the agent uses to login. <b>Note:</b> This overrides the logic of the place login.   |
| workmode           | This optional parameter specifies which workmode is applied to the AgentLogin request. Valid values are ManualIn and AutoIn.   |
| dynamicPhoneNumber | <p>This optional parameter is passed to the RequestAgentLogin operation in extensions. This parameter specifies the alternate phone number to which all SIP communication will be directed for the duration of the agent session. All agent-related and call-related events are generated for the DN for the default agent's place.</p> <p><b>Note:</b> SIP Server will reject the RequestAgentLogin operation if the alternate phone number provided by the agent already exists in the Configuration Server.</p> |

### Sample 1

#### Request

```
POST api/v2/me
{
  "operationName": "StartContactCenterSession",
  "channels": [
    "voice"
  ]
}
```

#### HTTP Response

```
{
  "statusCode": 0
}
```

#### CometD notification

This operation may not result in a notification. If the agent is logged in to a device as a result of starting the session, Web Services delivers a [DeviceStateChangeMessage](#) via CometD. If the agent is already logged in, Web Services skips the login request and doesn't return a notification.

### Sample 2

#### Request

```
POST api/v2/me
{
  "operationName": "StartContactCenterSession",
  "place": "SIP_5000",
  "loginCode": "8000",
  "queue": "9000",
  "channels": [
    "voice"
  ]
}
```

#### HTTP response

```
{
  "statusCode": 0
}
```

## Sample 3

### Request

```
POST /api/v2/me
{
  "operationName": "StartContactCenterSession",
  "channels": ["voice"],
  "place": "agent1",
  "loginCode": "agent1",
  "agentLoginPassword": "password"
}
```

### HTTP response

```
{
  "statusCode": 0
}
```

## Sample 4

The following example shows the partial success behavior:

### Request

```
POST /api/v2/me
{
  "channels": [
    "voice",
    "chat"
  ],
  "operationName": "StartContactCenterSession"
}
```

### HTTP response

```
{
  "failed": [
    {
      "channel": "multimedia",
      "statusCode": 3,
      "statusMessage": "Unsupported feature"
    }
  ],
  "statusCode": 7,
  "succeeded": [
    {
      "channel": "voice",
      "statusCode": 0
    }
  ]
}
```

### Sample 5

The following example shows workmode:

#### Request

```
POST /api/v2/me
{
  "operationName": "StartContactCenterSession",
  "workmode": "ManualIn",
  "channels": ["voice"],
  "place": "agent1@pizza.com"
}
```

#### HTTP response

```
{
  "statusCode": 0
}
```

The following example shows workmode with an invalid value:

#### Request

```
POST /api/v2/me
{
  "operationName": "StartContactCenterSession",
  "workmode": "aManualIn",
  "channels": ["voice"],
  "place": "agent1@pizza.com"
}
```

#### HTTP response

```
{
  "statusCode": 2,
  "statusMessage": "Parameter 'workmode' is invalid: \"The specified value is not within valid range\""
}
```

### Sample 6

The following example shows an alternate phone number:

#### Request

```
POST /api/v2/me
{
  "operationName": "StartContactCenterSession",
```

---

```
"channels":["voice"],
"dynamicPhoneNumber": "%dynamic_phone_number%"
}
```

### HTTP response

```
{
  "statusCode":0,
  "devices":[
    {
      "id":"b24aca59-5fda-4eac-88d2-dde979e5dc97",
      "deviceState":"Active",
      "userState":{"id":"2B36138D-
C564-4562-A8CB-3C32D564F296","displayName":"AuxWork","state":"NotReady","workMode":"AuxWork"},
      "phoneNumber":"%original_phone_number%",
      "description":"","",
      "e164Number":"","",
      "telephonyNetwork":"Private",
      "doNotDisturb":"Off",
      "voiceEnvironmentUri":"http://127.0.0.1/api/v2/voice-environments/
136ca479-fd97-42eb-9865-f792612955bc",
      "capabilities":["ForwardCallsOn"],
      "channels":["voice"],
      "dynamicPhoneNumber": "%dynamic_phone_number%"
    }
  ]
}
```