

# Agent Engagement API

## User Manual

# Table of Contents

Overview.....	5
API Token Authentication .....	6
Engagement Sequence .....	10
acceptEngagement (POST).....	12
acceptOwnership (POST).....	13
activePerformanceMetrics (GET).....	15
agentGroups (GET).....	16
agentStatus (POST) .....	17
autoTransferEngagement (POST).....	18
changePassword (POST) .....	20
closeEngagement (POST) .....	21
conferenceEngagement (POST) .....	26
conferenceMode (POST) .....	29
customScript (GET) .....	31
customScript (POST).....	33
customSettings (POST).....	35

denyEngagement (POST).....37

dispositions (GET).....38

escalateEngagement (POST) .....40

event/startDisposition (POST) .....41

extraChat (POST) .....42

followUpConversations (GET) .....43

followUpEngagement (POST).....45

fullScriptTree (GET).....47

joinEngagement (POST).....50

logging/agent (POST) .....52

login (POST) .....53

maskText (POST) .....55

messages (GET).....57

messages (POST) .....64

ninaCoachChatTakeOver (POST) .....66

ninaCoachContentMetaInfo (GET).....67

ninaCoachEscalationResolution (POST) .....69

**preLoginStatus (GET).....72**

**refuseOwnership (POST) .....73**

**requestEntireTranscripts (GET).....75**

**resolveConversation (POST) .....76**

**scriptGroup (GET).....81**

**scriptTree (GET).....83**

**sendingIllegalWordNotification (POST).....86**

**settings (GET) .....88**

**smsCommand (POST) .....91**

**summaryData (GET) .....93**

**transferConferenceOptions (GET).....97**

**transferEngagement (POST).....101**

**transferOwnership (POST).....104**

**verifySession (POST) .....106**

**webrtc/action (POST).....107**

# Overview

The Agent Engagement API (AEAPI) is used to power the Nuance Digital Engagement Platform (NDEP) Agent Desktop. The AEAPI also allows customers to build an alternative agent desktop or integrate their live chat program into an existing agent application. It can also be used to integrate a third-party virtual assistant into existing NDEP programs.

# API Token Authentication

Nuance Agent Engagement API (AEAPI) supports the OAuth2-based authorization framework for application-level tokens. OAuth2 provides authorization based on the Client Credentials Grant Flow of the OAuth2 specification.

Here is a list of the steps required to access the API using OAuth2-based authorization.

- Client registers their application with Nuance.
- Client application requests the access token using the Authorization Server.
- Client application uses the access token in every API request.

## Registering the Application

Clients may request application registration with Nuance by contacting their Client Services Manager (CSM). The CSM will register your application with the Nuance System Team.

The **clientId** is used by the service API to identify your application and to build login URLs.

The **clientSecret** is used to authenticate the identity of your application. It must be kept private between the application and the API.

---

**Note.** The client is responsible for storing the **clientId** and the **clientSecret** securely. If this information is compromised, you will have to request a lock on the account and request a new registration with your CSM.

---

## Obtaining an Access Token

The client application must connect to the Auth server using the `clientId` and `clientSecret` provided by Nuance. The `clientId` and `clientSecret` must be formatted properly for transmission. Formatting requires you to concatenate the `clientId`, a colon ":" character, and the `clientSecret` into a single string as follows:

```
clientId:clientSecret
```

The resulting string should be Base-64 encoded and inserted into the Authorization Header in the following format:

```
Authorization: Basic [Base-64 encoded string]
```

### Example

Request URL: <https://auth.touchcommerce.com/oauth-server/oauth/token>

Response Format: JSON

---

**Note.** Domain names may vary for each client.

---

### *Sample Request for Obtaining an Access Token*

POST <https://auth.touchcommerce.com/oauth-server/oauth/token>

#### Form Data

```
client_id:"ceapiId"  
grant_type:"client_credentials"
```

#### Form Data Details

- **client\_id** – The `clientId` provided by the CSM when the client application was registered with Nuance.
- **grant\_type** – This value should be `client_credentials` for the CE-API. When Nuance adds for other APIs, this value may be different for the new APIs.

#### *Request Header:*

```
Host: auth.touchcommerce.com  
Content-Type: application/x-www-form-urlencoded; charset=utf-8  
Authorization: Basic Y2VhcG1DbG11bnRjZDpjZWZwaUNsaWVudFNlY3JldA==
```

**Sample Response from Nuance**

```

{
  access_token:
    "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJhdWQiOiJ2VhcGkiXSwic2NvcGUiOiJ0b3R1eSIsInYyYyIj0sInNpdGVzIjpbIjE1NTM4LWlxYTknI3YTk1YiIsImNsaWVudF9pZCI6ImNlYXBpQ2xpZW50SWQifQ.jU3SaGeHYkxNaRVdbZhJfCI6rbeKjXQqfnjccib4gsChs9Ed2ZpbjgONpOUNdwJFfwE3BT3h-wDRiaSUqieczOXc058aIlyGuAw9nDDy4ex5sP9kGsOMekw3PJaIdYfqAKJW2BU9LJ1efvhPGNHZrV7ivOHVNHwacB-XeaGVVjbXXx4AJrHDrP-C-0fiK7AlCNuaYIGV47c9e3_479MGJpfEVcQVhhoZB6D1bVtrf3IFPLD456M5MoqC0kTgn2wJfmUSM77eCNDc7cMdDokvb1sAaOmhLMaJkuGfmvKXT7v7Yqm0_LA4GsHcPsmN9J8Dk4oyuhUTJYunBsdr_RRQ",
  token_type: "bearer",
  expires_in: 59,
  scope: "read write",
  sites : [ "154", "306" ],
  jti : "10dc19a9-48c4-4446-b141-d5179e5f21f1"
}

```

**Response Details**

- **Access token** – The access token must be included in all API data requests.
- **Token type** – The token type is “bearer” and must be included in all API data requests.
- **Expires in** – This value indicates the expiration time of the token in seconds. The client app should monitor this value and request a new token whenever necessary. If the app uses an expired token the server will return a code 401 “Access Token Expired” message.
- **Scope** – Scope indicates the access permissions for this token. If a disallowed action is performed, the server will return an error.
- **Sites** – Indicates the site IDs that are valid for this token. If the request tries to access a site that is not on the list, the server will return an error.
- **jti** – The ID for the access token that is used for tracking purposes. Clients can ignore this value.





# Engagement Sequence

## URI Format

Here is the basic format for an AE-API request:

```
https://api.touchcommerce.com/engagementAPI/{version}/{resource-type}/{resource}?{param_list}
```

For this request, “resource-type” is either ci, agent, or automaton.

## API Engagement Sequence

Operations for an API engagement to begin must be similar to the following:

1. Authentication
2. A POST, **login**, logs the agent in.
3. A GET, **message**, allows the system to retrieve the message until an agent accepts the engagement (not assigned to a specific agent.)
4. A POST, **acceptEngagement**, allows the agent to accept the engagement from the system so that the chat is now assigned to them.
5. A GET, **message**, allows the agent to retrieve the actual message, state, messageType etc.

After authentication, steps 2 through 4 must be done within 15 seconds. Otherwise the API will time out and show an error. Apps should periodically submit a GET message to keep the engagement active. Otherwise the system will timeout after approximately 30 seconds.

## Response Messages and Codes

Code	Reason
200	Request is successful.
201	Created.
400	Request has incorrect parameters.
401	There is no such agent on the server. (The agent needs to log in again or the engagement Id is incorrect, or the engagement cannot be created due to timeout.)
403	User is not authenticated to the API service.
404	Not found.
422	Request cannot be processed (maybe expired or inactivity or timeout).
500	Server error.

# acceptEngagement (POST)

This REST call confirms that the engagement has been accepted by the Agent Desktop.

URI	HTTP Method	Output Formats
acceptEngagement	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID of the agent.	string	body	no
engagementId	Unique ID of the engagement.	string	body	yes
registeredId	Id retrieved after agent login	string	query	yes

## Example Request

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/acceptEngagement?registeredId=123&output=json>

```
{
  "agentId": "string",
  "engagementId": "string"
}
```

## Response Class

Status 200

# acceptOwnership (POST)

This request indicates that the agent accepted the ownership transfer request. This request must be sent from the agent to whom the ownership transfer is made. The agent who initiated the transfer will be notified upon success of the transfer.

URI	HTTP Method	Output Formats
acceptOwnership	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID of the agent.	string	body	yes
engagementId	Unique ID of the engagement.	string	body	yes
reason	Reason for the transfer.	string	body	no
registeredId	Id retrieved after agent login	string	query	yes

## Example Request

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/acceptOwnership?registeredId=123&output=json>

```
{
  "agentId": "string",
  "engagementId": "string",
  "reason": "string"
}
```

**Response Class**

Status 200

```
{  
  "engagementId": "string",  
  "reason": "string",  
  "status": "string"  
}
```

# activePerformanceMetrics (GET)

Settings are retrieved on the AEAPI side.

URI	HTTP Method	Formats
activePerformanceMetrics	GET	XML or JSON

## Parameters

Name	Description	Data Type	Parameter Type	Required
agentId	Unique ID for the agent	string	query	yes
registeredId		string	query	yes
settingIds	Id for settings. One or comma-separated.	string	query	yes

## Example Request URL

<https://api.touchcommerce.com/engagementAPI/v2/agent/activePerformanceMetrics?registeredId=123&agentId=123&output=xml&settingIds=123>

## Response Class

Status 200

```
{
  "result": {},
  "setOrExpired": true
}
```

# agentGroups (GET)

This endpoint is used to fetch all agent groups available to an advanced agent prior to login.

URI	HTTP Method	Output Formats
agentGroups	GET	XML or JSON, JSON is default

## Parameters

The agentGroup endpoint does not accept parameters.

## Example Request

URL: <https://api.touchcommerce.com/engagementAPI/v2/agent/agentGroups?output=json>

## Response Class

Status 200

---

**Note.** The response delivers the agent's Agent Groups with names and IDs. The agentGroups endpoint supports the Agent Desktop "Advanced Agent" feature where an advanced agent is presented with a list of Agent Groups and IDs. This allows the agent to select the Agent Group they will take chats from.

---



# agentStatus (POST)

This REST call is used to post the agent status to the system.

URI	HTTP Method	Output Formats
agentStatus	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	body	yes
reason	Reason for posting the agent status.	string	body	no
state	available   busy	string	body	yes
registeredId	Id retrieved after agent login	string	query	yes

## Example Request

URL:

`https://api.touchcommerce.com/engagementAPI/v2/agent/agentStatus?registeredId=123&output=json`

```
{
  "agentId": "string",
  "reason": "string",
  "state": "available"
}
```

## Response Class

Status 200

# autoTransferEngagement (POST)

This REST call is used to post an auto transfer request to the server.

URI	HTTP Method	Output Formats
autoTransferEngagement	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentGroupId	The identification number of the agent group.	string	body	no
agentId	Unique ID for the agent.	string	body	yes
agentNotes	Notes written by the agent.	string	body	no
businessUnitId	Identification number of the business unit.	string	body	no
engagementId	Identification number of the engagement.	string	body	yes
registeredId	ID retrieved after agent login	string	query	yes

**Example Request**

URL:

`https://api.touchcommerce.com/engagementAPI/v2/agent/autoTransferEngagement?registeredId=123&ou  
tput=json`

```
{  
  "agentGroupId": "string",  
  "agentId": "string",  
  "agentNotes": "string",  
  "businessUnitId": "string",  
  "engagementId": "string"  
}
```

**Response Class**

Status 200

```
{  
  "engagementId": "string",  
  "reason": "string",  
  "status": "string"  
}
```

# changePassword (POST)

This REST call is used to change the agent's password.

URI	HTTP Method	Formats
changePassword	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	body	yes
confirmNewPassword	Confirm the new password.	string	body	yes
newPassword	New password.	string	body	yes
oldPassword	Expired password.	string	body	yes

## Example Request

URL: <https://api.touchcommerce.com/engagementAPI/v2/agent/changePassword?output=json>

```
{
  "agentId": "string",
  "confirmNewPassword": "string",
  "newPassword": "string",
  "oldPassword": "string"
}
```

## Response Class

Status 200

# closeEngagement (POST)

This method requests that the active engagement be closed by the agent. Disposition information is passed as part of this request in cases where disposition information is available.

URI	HTTP Method	Formats
closeEngagement	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentAttributes		array	body	no
agentGroupId		string	body	no
agentId		string	body	yes
agentNotes		string	body	no
autoTransferFailed	If the value is true, the auto transfer failed.	boolean	body	no
businessUnitId		string	body	no
dispositions	Dispositions information of an engagement.	dispositions	body	no
engagementId	Unique Id of the engagement.	string	body	yes
followUp	If the value is true, this is a follow-up action.	boolean	body	no
followUpAgent	An agent who will follow up the conversation. By default, if not set, it is the agentId.	string	body	no
followUpNotes	Notes added to the follow-up action.	string	body	no

Name	Value	Data Type	Parameter Type	Required
havaSessionId		string	body	no
ninaCoachTransfer		boolean	body	no
targetAgentId		string	body	no
transferConferenceMode	Indicates whether the chat is a transfer or in conference mode = ['transfer', 'conference']stringEnum:"transfer", "conference"	string	body	no
request-agent-attribute				
<ul style="list-style-type: none"> <li>name</li> </ul>		string	body	no
<ul style="list-style-type: none"> <li>value</li> </ul>		string	body	no
dispositions				
<ul style="list-style-type: none"> <li>categoryReasons</li> </ul>		array	body	
<ul style="list-style-type: none"> <li>dispositions</li> </ul>		disposition	body	no
<ul style="list-style-type: none"> <li>message</li> </ul>		string	body	no
disposition				
<ul style="list-style-type: none"> <li>category</li> </ul>	Disposition category.	string	body	no
<ul style="list-style-type: none"> <li>reasons</li> </ul>		array	body	no
disp				
<ul style="list-style-type: none"> <li>categories</li> </ul>		array	body	no
<ul style="list-style-type: none"> <li>scriptTree</li> </ul>		string	body	no

Name	Value	Data Type	Parameter Type	Required
<ul style="list-style-type: none"> <li>site</li> </ul>		string	body	no
dispositionCategory				
<ul style="list-style-type: none"> <li>id</li> </ul>		string	body	no
<ul style="list-style-type: none"> <li>name</li> </ul>		string	body	no
dispositionReason				
<ul style="list-style-type: none"> <li>id</li> </ul>		string	body	no
<ul style="list-style-type: none"> <li>name</li> </ul>		string	body	no
cat				
<ul style="list-style-type: none"> <li>label</li> </ul>		string	body	no
<ul style="list-style-type: none"> <li>reasons</li> </ul>		array	body	no
<ul style="list-style-type: none"> <li>value</li> </ul>		string	body	no
reason				
<ul style="list-style-type: none"> <li>label</li> </ul>		string	body	no
<ul style="list-style-type: none"> <li>value</li> </ul>		string	body	no

**Example Request URL**

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/closeEngagement?registeredId=123&output=json>

```
{
  "agentAttributes": [
    {
      "name": "string",
      "value": "string"
    }
  ],
  "agentGroupId": "string",
  "agentId": "string",
  "agentNotes": "string",
  "autoTransferFailed": false,
  "businessUnitId": "string",
  "dispositions": {
    "categoryReasons": [
      {
        "category": {
          "id": "string",
          "name": "string"
        },
        "reasons": [
          {
            "id": "string",
            "name": "string"
          }
        ]
      }
    ]
  },
  "dispositions": {
    "categories": [
      {
        "label": "string",
        "reasons": [
          {
            "label": "string",
            "value": "string"
          }
        ]
      }
    ]
  }
}
```



```
    ],  
    "value": "string"  
  }  
],  
"scriptTree": "string",  
"site": "string"  
},  
"message": "string"  
},  
"engagementId": "string",  
"followUp": false,  
"followUpAgent": "string",  
"followUpNotes": "string",  
"havaSessionId": "string",  
"ninaCoachTransfer": false,  
"targetAgentId": "string",  
"transferConferenceMode": "transfer"  
}
```

**Response Class**

Status 200

# conferenceEngagement (POST)

This method requests that the specified engagement be held in conference with the specified agent or agent group.

URI	HTTP Method	Output Formats
conferenceEngagement	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentAttributes	Used in cases where an agent attribute is needed for the conference. ["name", "value"]	array	body	no
<ul style="list-style-type: none"> <li>name</li> </ul>	Part of the <b>agentAttributes</b> array.	string		no
<ul style="list-style-type: none"> <li>value</li> </ul>	Part of the <b>agentAttributes</b> array.	string		no
agentGroupId	Unique ID for the agent group.	string	body	no
agentGroupName	Agent group name.	string	body	no
agentId	Unique ID for the agent.	string	body	yes
agentNotes	Optional notes from the transferring agent.	string	body	no
businessUnitId	Identification number of the business unit.	string	body	no
businessUnitName	Name of the business unit.	string	body	no
engagementId	Identification number of the engagement that will be held in conference.	string	body	yes

Name	Value	Data Type	Parameter Type	Required
prioritizeRequest	Request to prioritize the conference request ["true", "false"]	boolean	body	no
targetAgentId	Unique ID for the target agent.	string	body	no
registeredId	Id retrieved after agent login	string	query	yes

### Example Request

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/conferenceEngagement?registeredId=123&output=json>

```
{
  "agentAttributes": [
    {
      "name": "string",
      "value": "string"
    }
  ],
  "agentGroupId": "string",
  "agentGroupName": "string",
  "agentId": "string",
  "agentNotes": "string",
  "businessUnitId": "string",
  "businessUnitName": "string",
  "engagementId": "string",
  "prioritizeRequest": true,
  "targetAgentId": "string"
}
```

**Response Class**

Status 200

```
{  
  "engagementId": "string",  
  "reason": "string",  
  "status": "string"  
}
```

# conferenceMode (POST)

This endpoint is used to transition the agent who is a participant of the specified engagement from screen mode to full participant mode.

URI	HTTP Method	Formats
conferenceMode	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentAttributes	(Array[RequestAgentAttribute], <i>optional</i> ), ["name", "value"]	array	body	no
<ul style="list-style-type: none"> <li>name</li> </ul>	Part of <b>agentAttributes</b> array.			
<ul style="list-style-type: none"> <li>value</li> </ul>	Part of <b>agentAttributes</b> array.			
agentId	Unique ID for the agent.	string	body	yes
engagementId	Unique ID for the engagement that will be held in conference. Part of agentAttributes array	string	body	yes
registeredId	Id retrieved after agent login	string	query	yes

### Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/conferenceMode?registeredId=123&output=json
```

```
{
  "agentAttributes": [
    {
      "name": "string",
```

```
    "value": "string"  
  }  
],  
"agentId": "string",  
"engagementId": "string"  
}
```

**Response Class**

Status 200

# customScript (GET)

This call retrieves an agent custom script.

URI	HTTP Method	Formats
customScript	GET	XML or JSON

## Parameters

Name	Description	Data Type	Parameter Type	Required
agentId	Unique ID for the agent	string	query	

### Example Request URL

URL: <https://api.touchcommerce.com/engagementAPI/v2/agent/customScript?agentId=123&output=xml>

**Response Class**

Status 200

```
{
  "customScripts": {
    "lastUpdateDate": 0,
    "lng": "string",
    "scripts": [
      {
        "nonEditable": true,
        "scriptValue": "string",
        "type": "TEXT"
      }
    ]
  },
  "userName": "string"
}
```



# customScript (POST)

This REST call is used to change an agent custom script.

URI	HTTP Method	Output Formats
customScript	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
change-custom-script				
<ul style="list-style-type: none"> <li>agentId</li> </ul>	<b>Unique ID for the agent</b>			
<ul style="list-style-type: none"> <li>customScripts</li> </ul>		string	query	
CustomScripts				
<ul style="list-style-type: none"> <li>lastUpdateDate</li> </ul>		integer		no
<ul style="list-style-type: none"> <li>lng</li> </ul>		string		no
<ul style="list-style-type: none"> <li>scripts</li> </ul>		array		no
CustomScript				
<ul style="list-style-type: none"> <li>nonEditable</li> </ul>		boolean		no
<ul style="list-style-type: none"> <li>scriptValue</li> </ul>		string		no
<ul style="list-style-type: none"> <li>type</li> </ul>	['TEXT', 'HEADER']stringEnum:"TEXT", "HEADER"	string		no

**Example Request**

URL: <https://api.touchcommerce.com/engagementAPI/v2/agent/customScript?output=xml>

```
{
  "agentId": "string",
  "customScripts": {
    "lastUpdateDate": 0,
    "lng": "string",
    "scripts": [
      {
        "nonEditable": true,
        "scriptValue": "string",
        "type": "TEXT"
      }
    ]
  }
}
```

**Response Class**

Status 200

# customSettings (POST)

This REST call saves or changes an agent custom setting.

URI	HTTP Method	Formats
customSettings	POST	XML or JSON

## Parameters

Name	Description	Data Type	Parameter Type	Required
agentID	Unique agent ID.	string	query	yes
clicking	['singleClick', 'doubleClick']stringEnum:"singleClick", "doubleClick",	string		no
fontSize		integer		no
language		string		no
layoutSettings		string		no
loginState	[loginAsBusy', 'loginAsAvailable']stringEnum:"loginAsBusy", "loginAsAvailable",	string		no
playSounds		boolean		no
showAlerts		boolean		no
includeLayout		boolean	query	

**Example Request**

URL: <https://api.touchcommerce.com/engagementAPI/v2/agent/customSettings?output=json>

```
{  
  "agentId": "string",  
  "clicking": "singleClick",  
  "fontSize": 0,  
  "language": "en_US",  
  "layoutSettings": "string",  
  "loginState": "loginAsBusy",  
  "playSounds": false,  
  "showAlerts": false  
}
```

**Response Example Value**

Status 200

# denyEngagement (POST)

This REST call allows the Agent Desktop to deny an engagement. After receiving an engagement in a GET messages response, if the Agent Desktop is not ready to accept the engagement, it should explicitly deny using this endpoint. Otherwise, at the end of service, it is considered as a lost agent connection.

URI	HTTP Method	Formats
denyEngagement	POST	XML or JSON

## Parameters

Name	Value	Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	body	yes
engagementId	Unique ID for the engagement that will be denied.	string	body	yes
registeredId	Id retrieved after agent login.	string	query	yes

### Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/denyEngagement?registeredId=123&output=json
```

```
{
  "agentId": "string",
  "engagementId": "string"
}
```

### Response Example

Status 200

# dispositions (GET)

This element retrieves the list of available dispositions for the engagements. Dispositions are a list of predefined reasons for an agent to close an engagement. Although the client Agent Desktop can make this call at any time after the engagement is assigned to the agent, the suggested time to make this call is right before the agent decides to close the engagement.

URI	HTTP Method	Formats
dispositions	GET	XML or JSON

## Parameters

Name	Description	Data Type	Parameter Type	Required
username	Agent's user name.	string	query	yes
siteId	Unique site ID.	string	query	yes
scriptTreeId	Identification number of the script tree.	string	query	yes
registeredId	Id retrieved after agent login	string	query	yes

### Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/dispositions?registeredId=123&output=json&username=bob&siteId=123&scriptTreeId=123
```

**Response Class**

Status 200

```
{
  "categoryReasons": [
    {
      "category": {
        "id": "string",
        "name": "string"
      },
      "reasons": [
        {
          "id": "string",
          "name": "string"
        }
      ]
    }
  ],
  "dispositions": {
    "categories": [
      {
        "label": "string",
        "reasons": [
          {
            "label": "string",
            "value": "string"
          }
        ]
      },
      {
        "value": "string"
      }
    ],
    "scriptTree": "string",
    "site": "string"
  },
  "message": "string"
}
```

# escalateEngagement (POST)

This endpoint allows an agent to post escalation notes for the specified engagement.

URI	HTTP Method	Formats
escalateEngagement	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	body	yes
engagementId	Identification number of the engagement.	string	body	yes
escalateReason	Comments supplied by the agent at the time of escalation.	string	body	no
registeredId	Id retrieved after agent login	string	query	yes

### Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/escalateEngagement?registeredId=123&output=json
```

```
{
  "agentId": "string",
  "engagementId": "string",
  "escalateReason": "string"
}
```

### Response Class

Status 200



# event/startDisposition (POST)

This endpoint is used to fire a disposition event.

URI	HTTP Method	Formats
event/startDisposition	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	body	
engagementId	Unique ID for the engagement.	string	body	
registeredId	Id retrieved after agent login	string	query	yes

## Example Request URL

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/event/startDisposition?registeredId=123&output=json
```

```
{
  "agentId": "string",
  "engagementId": "string"
}
```

## Response Class

Status 200

# extraChat (POST)

This endpoint requests additional chat features

URI	HTTP Method	Formats
extraChat	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	UniqueID for the agent	string	body	
enabled	True if an additional chat is needed	boolean	body	
registeredId	Id retrieved after agent login	string	query	yes

### Example Request

URL `https://api.touchcommerce.com/engagementAPI/v2/agent/extraChat?registeredId=123&output=json`

```
{
  "agentId": "string",
  "enabled": false
}
```

### Response Class

Status 200

# followUpConversations (GET)

This endpoint retrieves all conversations where the agent uses the “follow-up” feature.

URI	HTTP Method	Formats
followUpConversations	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
username	Agent's username.	string	query	yes
registeredId	Id associated with the session.	string	query	yes

### Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/followUpConversations?output=json&username=bob&registeredId=123
```

**Response Class**

Status 200

```
{
  "conversations": [
    {
      "agentGroupId": "string",
      "businessRuleId": "string",
      "businessUnitId": "string",
      "chatTitle": "string",
      "conversationChannel": "string",
      "conversationId": "string",
      "customerId": "string",
      "followUpNotes": "string",
      "followUpTimeout": 0,
      "followUpTimestamp": 0,
      "lastUseTimestamp": 0,
      "launchPage": "string",
      "launchType": "string",
      "siteId": "string"
    }
  ]
}
```

# followUpEngagement (POST)

This endpoint initiates an engagement for the selected follow-up conversation.

URI	HTTP Method	Formats
followUpEngagement	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentGroupID	Id for the agent group that owns the follow-up engagement.	string	body	no
agentId	Unique ID of the agent.	string	query	yes
businessRuleId	Identification number of the business rule.	string	body	no
businessUnitId	Identification number of the business unit.	string	body	no
chatTitle	Title of the chat.	string	body	no
conversationChannel	Channel used to transmit the conversation.	string	body	no
conversationId	Identification number of the conversation.	string	body	no
customerId	Identification number of the customer.	string	body	no
launchPage	Web page that launched the engagement.	string	body	no
launchType		string	body	no
siteId	Unique site ID	string	body	no

Name	Value	Data Type	Parameter Type	Required
registeredId	Id retrieved after agent login	string	query	yes

**Example Request**

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/followUpEngagement?registeredId=123&output=json
```

```
{  
  "agentGroupId": "string",  
  "agentId": "string",  
  "businessRuleId": "string",  
  "businessUnitId": "string",  
  "chatTitle": "string",  
  "conversationChannel": "string",  
  "conversationId": "string",  
  "customerId": "string",  
  "launchPage": "string",  
  "launchType": "string",  
  "siteId": "string"  
}
```

**Response Class**

Status 200

# fullScriptTree (GET)

This endpoint retrieves from the server the full script including all script groups associated with a specific site ID.

URI	HTTP Method	Formats
followUpConversations	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
registeredId	Id retrieved after agent login	string	query	yes
username	Agent's username.	string	query	
siteId	Site Id	string	query	

## Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/fullScriptTree?registeredId=123&output=json&username=qwe&siteId=123
```

## Response Class

Status 200

```
{
  "scriptTreesOfSite": [
    {
      "checksum": "string",
      "scriptTrees": [
        {
          "id": "string",
```

```

"name": "string",
"scriptGroupFolder": [
  {
    "name": "string",
    "scriptGroup": [
      {
        "errorMsg": "string",
        "id": "string",
        "name": "string",
        "scriptGroup": [
          {
            "automaton-id": "string",
            "automaton-type": "string",
            "command": "string",
            "devices": "string",
            "hotkey": "string",
            "keywords": "string",
            "nonEditable": "string",
            "scriptType": "string",
            "scriptValue": "string"
          }
        ]
      }
    ],
    "scriptGroupFolder": [
      {}
    ]
  }
],
"scriptGroups": [
  {
    "errorMsg": "string",
    "id": "string",
    "name": "string",
    "scriptGroup": [
      {
        "automaton-id": "string",
        "automaton-type": "string",
        "command": "string",

```



```
        "devices": "string",
        "hotkey": "string",
        "keywords": "string",
        "nonEditable": "string",
        "scriptType": "string",
        "scriptValue": "string"
    }
  ]
}
],
"siteId": "string"
}
]
```

# joinEngagement (POST)

This endpoint is used to enter an engagement. This action is used in cases where the agent has an engagement but then loses the connection. Once the connection is restored, the agent can send this request to establish a connection with the engagement again. (Do not use this method to accept a new engagement request. Use acceptEngagement instead.)

URI	HTTP Method	Formats
joinEngagement	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	body	yes
engagementId	Unique ID for an engagement.	string	body	yes
messageText		string		no
status		string		no
registeredId	Id retrieved after agent login	string	query	yes

### Example Request

URL:

`https://api.touchcommerce.com/engagementAPI/v2/agent/joinEngagement?registeredId=qwe&output=json`

**Response Class**

Status 200

```
{  
  "agentId": "string",  
  "engagementId": "string",  
  "messageText": "string",  
  "status": "string"  
}
```

# logging/agent (POST)

This REST call is used to save error messages in the log file.

URI	HTTP Method	Formats
logging/agent	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	body	yes
engagementId	Unique ID for an engagement.	string	body	yes
registeredId	Id retrieved after agent login	string	query	yes

### Example Request URL

URL: `https://api.touchcommerce.com/engagementAPI/v2/agent/logging/agent`

### Response Class

Status 200

# login (POST)

This REST call authenticates an agent and adds them to the list of available agents that can receive new engagements. In response, it returns the agent ID, status, and agent-specific information.

URI	HTTP Method	Formats
login	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
password	Agent's password.	string	body	yes
username	Agent's username.	string	body	yes

## Example Request

URL: <https://api.touchcommerce.com/engagementAPI/v2/agent/login?output=json>

```
{  
  "password": "string",  
  "username": "string"  
}
```

**Response Class**

Status 200

```
{  
  "loginAsAvailable": true,  
  "publicUserId": "string",  
  "registeredId": "string",  
  "status": "success",  
  "userId": "string"  
}
```

# maskText (POST)

This endpoint indicates which text in the chat line will be masked in the database.

URI	HTTP Method	Formats
maskTestRequest	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique Id of the agent.	string	body	
engagementId	Unique Id of the agent	string	body	
maskedChatLine	Chat line for which masking is requested	string	body	
previousMaskedChatLine	The chat line before the requested masking. This may contain previous masking.	string	body	
registeredId	Id retrieved after agent login.	string	query	yes

### Example Request URL

URL: <https://api.touchcommerce.com/engagementAPI/v2/agent/maskText?registeredId=123&output=json>

```
{
  "agentId": "string",
  "engagementId": "string",
  "maskedChatLine": "string",
  "previousMaskedChatLine": "string"
}
```

**Response Class**  
Status 200



# messages (GET)

This endpoint retrieves status updates for each active engagement. It also collects pending customer messages. The Agent Desktop client can configure their requests to either pend on the server and wait for new messages or return immediately and poll the endpoint periodically for updates.

Example responses include:

- Nothing pending – a 204 HTTP status code is returned.
- One or more new customer messages with corresponding chatID for each message.

URI	HTTP Method	Formats
messages	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter	Required
agentId	Agent's user name.	query	string	
instantResponse	True or false. If true, then return response immediately without waiting	query	boolean	no
requestEntireTranscript	True or false. If true, the entire transcript for the specified engagement will be returned in a single response.	query	boolean	no

### Example Response

<http://aeapi.touchcommerce.com:10080/engagementAPI/v2/agent/swagger-ui.html#!/agent-eapi-controller/getMessagesUsingGET>

## Response Values

Name	Value	Type	Required
engagementID	<<engagementID>>	string	Wrapper for messages related to the engagement
chatLineReceiverType	external internal	string	chatLineReceiverType=external, if received from customer.  chatLineReceiverType=internal, if received from agent or supervisor.
conferenceMode	Full   screen	string	The conference mode of the engagement.  Required only if the chat request type is conference.
isTransferOwnership	true   false	string	Required only if the chat request type is conference.
<settings>	<settings>	string	wrapper around the Engagement Specific settings.
<EngagementSettings>	<setting>		Each Engagement specific settings. Please look into (GET)messages Response Model in Swagger to understand the data returned as part of each engagement.
messages	<messages>	string	Wrapper for all messages

Name	Value	Type	Required
messageType	chatLine, agentDataPass, scriptLine, stateChange, newRequest, customerData, opener, automatonRequest, automatonResponse, xform, system, maskText, settingChanged, error, cobrowse, smsStatus, agentKicked, ninaAnswer	string	<p>Type of message that is being returned. When the messageType is newRequest, 'requestType' field information also passed.</p> <p>When the "stateChange" message type is received, the "state" field information also passed indicating what state was changed.</p> <p>When the message type is "error" and messageText is "orig.acknowledge.absent" the current chat should be closed on agent application side.</p> <p>Examples of System messages are : 'Agent A1 Forced Logged out', 'Agent A1 re-logins' , 'Agent is disconnected because of three missed attempts to accept OR decline'</p>

Name	Value	Type	Required
state	customerClosedEngagement, agentClosedEngagement, agentExitedFromEngagement, customerLostConnection, agentLostConnection, agentConnected, agentReconnected, customerStartsTyping, customerStopsTyping, customerMinimizedChatWindow, customerRestoredChatWindow, customerNavigatedToPage, startedMoveToPersistent, movedToPersistent, saleOccurred, ownershipTransferRequest, ownershipChange, supervisorConnected, supervisorLostConnection, supervisorExitedFromEngagement, customerClosedEngagement, agentClosedEngagement, agentExitedFromEngagement, customerLostConnection, agentLostConnection, agentConnected, agentReconnected, customerStartsTyping, customerStopsTyping, customerMinimizedChatWindow, customerRestoredChatWindow, customerNavigatedToPage, startedMoveToPersistent, movedToPersistent, saleOccurred, ownershipTransferRequest, ownershipChange, supervisorConnected, supervisorLostConnection, supervisorExitedFromEngagement	string	The current status of the engagement. This is required when the message type is stateChange.
message	Tag to wrap the content of each message	string	Indicates that one or more customer messages are included in the response.
messageData	{"key":"value"} response for rich data to be passed back to agent. Note: string should be encoded in the request e.g. %7B%22key%22%3A%22value%22%7D	string	no
messageText	Chat line text	string	The actual text/message.
requestType	transfer   conference   engagement autoTransfer	string	Required when message type is newRequest.

Name	Value	Type	Required
customerData	<<customerData>>		Information about customer. For example, OS, page market, browser, etc.
pageMarker	<<pageMarker into>>		

**Example Request**

GET https://api.touchcommerce.com/engagementAPI/v1/agent/messages?agentID=123456789

**Example Value**

```
<?xml version="1.0"?>
<EngagementMessagesResponse>
  <engagements>
    <id>string</id>
    <messages>
      <chatLineReceiverType>external</chatLineReceiverType>
      <customerDataList>
        <data>string</data>
        <name>string</name>
      </customerDataList>
      <member>agent</member>
      <messageText>string</messageText>
      <messageType>chatLine</messageType>
      <pageMarker>string</pageMarker>
      <reassignment>true</reassignment>
      <requestType>engagement</requestType>
      <state>customerClosedEngagement</state>
      <transcript>
        <senderId>string</senderId>
        <senderName>string</senderName>
        <text>string</text>
        <timestamp>string</timestamp>
        <type>string</type>
      </transcript>
      <url>string</url>
      <userId>string</userId>
    </messages>
  </engagements>
</EngagementMessagesResponse>
```

```

</messages>
<settings>
  <agentAlias>string</agentAlias>
  <agentGroupId>string</agentGroupId>
  <agentGroupName>string</agentGroupName>
  <autoOwnershipTransfer>true</autoOwnershipTransfer>
  <autoTransfer>true</autoTransfer>
  <autoTransferTimeout>string</autoTransferTimeout>
  <autoTransferWarnTimeout>string</autoTransferWarnTimeout>
  <automaton>true</automaton>
  <businessRuleId>string</businessRuleId>
  <businessUnitId>string</businessUnitId>
  <businessUnitName>string</businessUnitName>
  <callEnabled>true</callEnabled>
  <callInformation>string</callInformation>
  <callNeeded>true</callNeeded>
  <chatTitle>string</chatTitle>
  <clientId>string</clientId>
  <clientName>string</clientName>
  <deviceType>string</deviceType>
  <dispositions>true</dispositions>
  <forced>true</forced>
  <joinInScreeningMode>true</joinInScreeningMode>
  <language>string</language>
  <launchPage>string</launchPage>
  <launchType>string</launchType>
  <manualJoinConference>true</manualJoinConference>
  <multipleDispositions>true</multipleDispositions>
  <persistent>true</persistent>
  <scriptTreeChecksum>string</scriptTreeChecksum>
  <scriptTreeId>string</scriptTreeId>
  <siteId>string</siteId>
</settings>
</engagements>
<messages>
  <chatLineReceiverType>external</chatLineReceiverType>
  <customerDataList>
    <data>string</data>
    <name>string</name>

```

```
</customerDataList>  
<member>agent</member>  
<messageText>string</messageText>  
<messageType>chatLine</messageType>  
<pageMarker>string</pageMarker>  
<reassignment>true</reassignment>  
<requestType>engagement</requestType>  
<state>customerClosedEngagement</state>  
<transcript>  
  <senderId>string</senderId>  
  <senderName>string</senderName>  
  <text>string</text>  
  <timestamp>string</timestamp>  
  <type>string</type>  
</transcript>  
<url>string</url>  
<userId>string</userId>  
</messages>  
</EngagementMessagesResponse>
```

# messages (POST)

This REST call is used to post a message to the specified engagement.

Automatons passed through **messageText** on the POST call are not supported. Here is an example for agent-pushed (automatons) forms.

```
requestBody {
  "agentId": "jnajarro1",
  "chatLineReceiverType": "external",
  "engagementId": "9098777111835804841",
  "messageText": "<span data-automaton-id=\"2046\" data-context=\"ci\" data-panel=\"chat-footer\" data-action=\"show\">Requested Credit Card Payment form</span>",
  "messageType": "script",
  "scriptTreeId": "200118",
  "scriptType": "text",
  "state": "agentIsTyping"
}
```

Parameter content type:

URI	HTTP Method	Formats
messages	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID of the agent.	string	body	yes
chatLineReceiverType	external   internal	string	body	no
command	String(optional) send automatons built in xforms	string	body	no



Name	Value	Data Type	Parameter Type	Required
description	Description of the message (optional).	string	body	no
engagementId	Unique ID of the engagement.	string	body	yes
messageData	Rich content data to be rendered to the customer, This field is used to pass any payload more complex than a simple string such as jsonString, jsonObject, HTML tage, etc.	string	body	no
messageText	Sends any String to the customer (send automatons built using acif). To pass a payload more complex than a simple string such as jsonObject, jsonString, div tags (etc.) use the messageData field instead.	string	body	no
messageType	chatLine   stateChange   script	string	body	no
scriptTreeld	Id of the script tree that the script belongs to if specified	string	body	no
scriptType	text   command <b>Note.</b> The scriptType parameter allows you to send script commands when the messageType parameter is set to "script."	string	body	no
state	agentsTyping   agentStopsTyping	string	body	no
registeredId	Id retrieved after agent login	string	query	yes

**Request URL**

URL: <https://api.touchcommerce.com/engagementAPI/v2/agent/messages?output=json&registeredId=123>

# ninaCoachChatTakeOver (POST)

This REST call is used to get ownership of the chat to the current live agent from a virtual agent.

URI	HTTP Method	Formats
ninaCoachChatTakeOver	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.(target agent)	string	body	yes
engagementId	Unique ID the engagement	string	body	yes
havaSessionId	Id for the virtual agent session	string	body	no
registeredId	Id retrieved after agent login	string	query	yes

### Example Request URL

URL:

`https://api.touchcommerce.com/engagementAPI/v2/agent/ninaCoachChatTakeOver?registeredId=123&output=json`

```
{
  "agentId": "string",
  "engagementId": "string",
  "havaSessionId": "string"
}
```

### Response Class

Status 200

# ninaCoachContentMetaInfo (GET)

This REST call is used to receive meta info about current chat from remote Nina server.

URI	HTTP Method	Formats
ninaCoachContentMetaInfo	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	query	
contentVersion	Property previously parsed from the new engagement message (ninaCoachConversation field). Used by remote Nina server.	string	query	
engagementId	Unique ID for the engagement.	string	query	
havaSessionId	Unique Id generated by the ChatRouter for current chat escalation (copied from new engagement message). Used by Nuance servers for logging events.	string	query	
registeredId	registeredId	string	query	yes
sci	Property previously parsed from the new engagement message (ninaCoachConversion field). Used by remote Nina server.	string	query	

**Example Request**

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/ninaCoachContentMetaInfo?registeredId=123&engagementId=123&havaSessionId=123&contentVersion=2&sci=123&output=json
```

**Response Class**

Status 200

```
{  
  "contentMetaInfo": "string",  
  "contentVersion": "string"  
}
```

# ninaCoachEscalationResolution (POST)

This REST call is used to inform the virtual agent with intent and concepts selected by the live agent if the live agent was able to help.

URI	HTTP Method	Formats
ninaCoachEscalationResolution	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
ninaCoachEscalationResolution	Unique ID for the agent.(target agent)	string	body	yes
• activeIntent		string		no
• activeIntentId		string		no
• activeIntentText		string		no
• agentId	Unique agent Id.	string		
• concepts				no
• engagementId		string		
• havaSessionId		string		
• iid		string		
• intent		string		no

Name	Value	Data Type	Parameter Type	Required
• intentId		string		no
• intentText		string		no
• resolutionCode	(string) = ['NL', 'ERR', 'TIMEOUT', 'NOOP', 'INC', 'LC']stringEnum:"NL", "ERR", "TIMEOUT", "NOOP", "INC", "LC",	string		
• sci		string		
• status		string		no
• concept				
○ name		string		
○ value		string		
registeredId		string	query	yes

**Example Request**

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/ninaCoachEscalationResolution?registerId=123&output=json>

```
{
  "activeIntent": "string",
  "activeIntentId": "string",
  "activeIntentText": "string",
  "agentId": "string",
  "concepts": [
    {
      "name": "string",
      "value": "string"
    }
  ],
  "engagementId": "string",
  "havaSessionId": "string",
  "iid": "string",
  "intent": "string",
  "intentId": "string",
  "intentText": "string",
  "resolutionCode": "NL",
  "sci": "string",
  "status": "string"
}
```

**Response Class**

Status 200

# preLoginStatus (GET)

Retrieves the information required before login action (such as whether the agent is already logged in). No JWT token is required.

URI	HTTP Method	Formats
preLoginStatus	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
username	Agent's username.	string	query	

## Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/preLoginStatus?output=json&username=bob
```

## Response Class

Status 200

```
{  
  "hasActiveSession": true  
}
```



# refuseOwnership (POST)

This request indicates that the target agent refused the request for transfer for engagement ownership. This request must be sent from the agent to whom ownership transfer request was sent. The initiating transfer agent will be notified upon successful rejection of ownership.

URI	HTTP Method	Formats
refuseOwnership	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.(target agent)	string	body	yes
engagementId	Unique ID the engagement	string	body	yes
reason	Reason for the refusal of ownership.	string	body	no
registeredId	Id retrieved after agent login	string	query	yes

**Example Request URL**

URL:

`https://api.touchcommerce.com/engagementAPI/v2/agent/refuseOwnership?registeredId=9876876876&output=json`

```
{  
  "agentId": "string",  
  "engagementId": "string",  
  "reason": "string"  
}
```

**Response Class**

Status 200

# requestEntireTranscripts (GET)

Used to get one or more chat transcripts.

URI	HTTP Method	Formats
requestEntireTranscripts	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
username	The agent's username	string	query	yes
chatIds	Chat IDs of the requested transcripts. One value can be listed or multiple values can be written using a comma separator.	string	query	yes
registeredId	Id retrieved after agent login.	string	query	yes

### Example Request URL

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/requestEntireTranscripts?registeredId=3456356&username=zaher&chatIds=23&output=xml
```

### Response Class

Status 200

# resolveConversation (POST)

Requests that the conversation be resolved by an agent. If disposition information exists, it is also passed as part of this request.

URI	HTTP Method	Formats
logout	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
resolveConversationRequest				
<ul style="list-style-type: none"> <li>agentAttributes</li> </ul>		array		no
<ul style="list-style-type: none"> <li>agentGroupId</li> </ul>	Agent group Id.	string		no
<ul style="list-style-type: none"> <li>agentId</li> </ul>	Unique Id of the agent being logged out.	string	body	yes
<ul style="list-style-type: none"> <li>agentNotes</li> </ul>	Notes entered by the agent.	string		no
<ul style="list-style-type: none"> <li>autoTransferFailed</li> </ul>	This value is true if the auto transfer failed.	boolean	body	no
<ul style="list-style-type: none"> <li>businessUnitId</li> </ul>	Id of the business unit.	string		no
<ul style="list-style-type: none"> <li>conversationId</li> </ul>	Unique Id of the conversation.	string	body	no
<ul style="list-style-type: none"> <li>dispositions</li> </ul>	Disposition information about the engagement.	string	body	no
<ul style="list-style-type: none"> <li>engagementId</li> </ul>	Unique engagement Id.	string	body	yes
<ul style="list-style-type: none"> <li>followUp</li> </ul>	Follow-up action.	boolean	body	no

Name	Value	Data Type	Parameter Type	Required
<ul style="list-style-type: none"> <li>followUpAgent</li> </ul>	Agent that will follow up the conversation. Used by default. Otherwise agentId can be used.	string	body	no
<ul style="list-style-type: none"> <li>followUpNotes</li> </ul>	Contains notes added to the follow-up action.	string	body	no
<ul style="list-style-type: none"> <li>havaSessionId</li> </ul>		string		no
<ul style="list-style-type: none"> <li>ninaCoachTransfer</li> </ul>		boolean		no
<ul style="list-style-type: none"> <li>targetAgentId</li> </ul>		string		no
<ul style="list-style-type: none"> <li>transferConferenceMode</li> </ul>	Indicates whether transfer or conference mode is used.	string	body	no
request-agent-attribute				
<ul style="list-style-type: none"> <li>name</li> </ul>		string		no
<ul style="list-style-type: none"> <li>value</li> </ul>		string		no
dispositions				
<ul style="list-style-type: none"> <li>categoryReasons</li> </ul>		array		no
<ul style="list-style-type: none"> <li>dispositions</li> </ul>				no
<ul style="list-style-type: none"> <li>message</li> </ul>		string		no
disposition				
<ul style="list-style-type: none"> <li>category</li> </ul>	Disposition category.			
<ul style="list-style-type: none"> <li>reasons</li> </ul>		array		no
disp				
<ul style="list-style-type: none"> <li>categories</li> </ul>		array		no
<ul style="list-style-type: none"> <li>scriptTree</li> </ul>		string		no

Name	Value	Data Type	Parameter Type	Required
<ul style="list-style-type: none"> <li>site</li> </ul>		string		no
dispositionCategory				
<ul style="list-style-type: none"> <li>id</li> </ul>		string		no
<ul style="list-style-type: none"> <li>name</li> </ul>		string		no
dispositionReason				
<ul style="list-style-type: none"> <li>id</li> </ul>		string		no
<ul style="list-style-type: none"> <li>name</li> </ul>		string		no
cat				
<ul style="list-style-type: none"> <li>label</li> </ul>		string		no
<ul style="list-style-type: none"> <li>reasons</li> </ul>		array		no
<ul style="list-style-type: none"> <li>value</li> </ul>		string		no
reason				
<ul style="list-style-type: none"> <li>label</li> </ul>		string		no
<ul style="list-style-type: none"> <li>value</li> </ul>		string		no

**Example Request**

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/resolveConversation?registeredId=123&output=json>

```
{
  "agentAttributes": [
    {
      "name": "string",
      "value": "string"
    }
  ]
}
```



```
    }  
  ],  
  "agentGroupId": "string",  
  "agentId": "string",  
  "agentNotes": "string",  
  "autoTransferFailed": false,  
  "businessUnitId": "string",  
  "conversationId": "string",  
  "dispositions": {  
    "categoryReasons": [  
      {  
        "category": {  
          "id": "string",  
          "name": "string"  
        },  
        "reasons": [  
          {  
            "id": "string",  
            "name": "string"  
          }  
        ]  
      }  
    ],  
    "dispositions": {  
      "categories": [  
        {  
          "label": "string",  
          "reasons": [  
            {  
              "label": "string",  
              "value": "string"  
            }  
          ]  
        }  
      ],  
    }  
  ],  
}
```

```
        "value": "string"
      }
    ],
    "scriptTree": "string",
    "site": "string"
  },
  "message": "string"
},
"engagementId": "string",
"followUp": false,
"followUpAgent": "string",
"followUpNotes": "string",
"havaSessionId": "string",
"ninaCoachTransfer": false,
"targetAgentId": "string",
"transferConferenceMode": "transfer"
}
```

**Response Class**

Status 200

```
{
  "resolveState": "OK",
  "warnParams": "string",
  "warning": "string"
}
```



# scriptGroup (GET)

This REST call retrieves the scripts associated with the specified script group ID from the server.

URI	HTTP Method	Response Format
scriptGroup	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
username	Agent's username	string	query	yes
scriptGroupId	The ID of the script group for which this call will retrieve the scripts.	string	query	yes

### Example Request URL

`https://api.touchcommerce.com/engagementAPI/v2/agent/scriptGroup?registeredId=3564563465&output=json&username=zaher&scriptGroupId=23`

**Response Class**

Status 200

```
{
  "errorMsg": "string",
  "id": "string",
  "name": "string",
  "scriptGroup": [
    {
      "automaton-id": "string",
      "automaton-type": "string",
      "command": "string",
      "devices": "string",
      "hotkey": "string",
      "keywords": "string",
      "nonEditable": "string",
      "scriptType": "string",
      "scriptValue": "string"
    }
  ]
}
```

# scriptTree (GET)

Retrieves the script tree associated with a specific site ID from the server.

URI	HTTP Method	Response Format
scriptTree	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
username	Agent's username.	string	query	yes
siteId	The ID of the script group for which this call will retrieve the scripts.	string	query	no
registeredId	Id retrieved after agent login	string	query	yes

### Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/scriptTree?registeredId=9868769876&output=json&username=zaher&siteId=23
```

**Response Class**

Status 200

```

{
  "scriptTreesOfSite": [
    {
      "checksum": "string",
      "scriptTrees": [
        {
          "id": "string",
          "name": "string",
          "scriptGroupFolder": [
            {
              "name": "string",
              "scriptGroup": [
                {
                  "errorMsg": "string",
                  "id": "string",
                  "name": "string",
                  "scriptGroup": [
                    {
                      "automaton-id": "string",
                      "automaton-type": "string",
                      "command": "string",
                      "devices": "string",
                      "hotkey": "string",
                      "keywords": "string",
                      "nonEditable": "string",
                      "scriptType": "string",
                      "scriptValue": "string"
                    }
                  ]
                }
              ]
            }
          ],
          "scriptGroupFolder": [
            {}
          ]
        }
      ],
      "scriptGroups": [

```

```
{
  "errorMsg": "string",
  "id": "string",
  "name": "string",
  "scriptGroup": [
    {
      "automaton-id": "string",
      "automaton-type": "string",
      "command": "string",
      "devices": "string",
      "hotkey": "string",
      "keywords": "string",
      "nonEditable": "string",
      "scriptType": "string",
      "scriptValue": "string"
    }
  ]
},
],
"siteId": "string"
}
]
```

# sendIllegalWordNotification (POST)

This endpoint is used to send illegal word notifications.

URI	HTTP Method	Formats
sendIllegalWordNotification	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
illegal-word-notification				
<ul style="list-style-type: none"> <li>agentId</li> </ul>	Unique Id for the agent.	string	body	
<ul style="list-style-type: none"> <li>engagementId</li> </ul>	Unique Id for the engagement	string	body	
<ul style="list-style-type: none"> <li>illegalWord</li> </ul>	The illegal word that was used and that caused the notification.	string	body	
<ul style="list-style-type: none"> <li>settingIds</li> </ul>	Notification settings	array	body	
<ul style="list-style-type: none"> <li>transcriptLine</li> </ul>	Text line where the illegal word was used.	string	body	
registered	Id retrieved after agent login	string	query	

**Example Request**

URL:

`https://api.touchcommerce.com/engagementAPI/v2/agent/sendIllegalWordNotification?registeredId=123`

```
{
  "agentId": "string",
  "engagementId": "string",
  "illegalWord": "string",
  "settingIds": [
    0
  ],
  "transcriptLine": "string"
}
```

**Response Class**

Status 200

# settings (GET)

Retrieves global settings for specific group of types.

URI	HTTP Method	Formats
settings	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
output	Response format.	string	query	
username	Agent's username.	string	query	
registeredId	Id associated with the session.	string	query	yes
settingTypes	Comma separated setting types.	string	query	

### Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/settings?output=json&username=bob&registeredId=123&settingTypes=123
```



**Response Class**

Status 200

```
{
  "busyStatuses": {
    "status": [
      "string"
    ]
  },
  "cobrowseEnabled": true,
  "cobrowseURLs": [
    {
      "siteId": "string",
      "url": "string"
    }
  ],
  "enableAudibleAlert": true,
  "enableAudibleAlertWhenNotInFocus": true,
  "enableCustomScript": true,
  "enableMoveWindow": true,
  "loginStatus": "available",
  "maxChats": 0,
  "maxExtraSlots": 0,
  "maxForcedChats": 0,
  "reservedStatuses": {
    "status": [
      "string"
    ]
  },
  "screenCaptureLog": true,
  "siteSettings": {
    "sites": [
      {
        "agentGroups": [
          {
            "displayName": "string",
            "id": "string",
            "name": "string",
            "settingId": "string"
          }
        ]
      }
    ]
  }
}
```

```
    ],  
    "businessUnits": [  
      {  
        "displayName": "string",  
        "id": "string",  
        "name": "string",  
        "settingId": "string"  
      }  
    ],  
    "id": "string",  
    "name": "string"  
  }  
]  
},  
"visualIndicatorFlashing": "NONE"  
}
```

# smsCommand (POST)

This endpoint is used to send the engagement command.

URI	HTTP Method	Formats
smsCommand	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
sms-chat-command				
<ul style="list-style-type: none"> <li>agentId</li> </ul>	Unique Id for the agent.	string	body	
<ul style="list-style-type: none"> <li>command</li> </ul>	['RESEND', 'SKIP'] stringEnum:"RESEND", "SKIP",	string	body	
<ul style="list-style-type: none"> <li>engagementId</li> </ul>	Unique Id for the engagement	string	body	
<ul style="list-style-type: none"> <li>lineId</li> </ul>	Notification settings	string	body	
registered	Id retrieved after agent login	string	query	

### Example Request

URL:

`https://api.touchcommerce.com/engagementAPI/v2/agent/smsCommand?registeredId=123&output=json`

```
{
  "cache-control": "no-store",
  "content-encoding": "gzip",
  "content-type": "application/xml",
  "date": "Wed, 17 Apr 2019 23:08:25 GMT",
  "pragma": "no-cache",
  "transfer-encoding": "chunked",
  "vary": "Accept-Encoding",
  "www-authenticate": "Bearer realm=\"aeapi\", error=\"unauthorized\", error_description=\"Full authentication is required to access this resource\"",
  "x-content-type-options": "nosniff",
  "x-frame-options": "DENY",
  "x-xss-protection": "1; mode=block"
}
```

### Response Code

401

# summaryData (GET)

This endpoint is used to get an agent's summary data about total logged in agents, engagements in queue, and available slots to get engagement.

URI	HTTP Method	Formats
summaryData	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	query	yes
registeredId	Id retrieved after agent login	string	query	yes

## Example Request

URL:

```
https://api.touchcommerce.com/engagementAPI/v2/agent/summaryData?registeredId=123&agentId=123&output=json
```

## Response Class

Status 200

```
{
  "businessUnits": [
    {
      "agentAttributes": [
        {
          "agents": [
            {
              "availableSlots": 0,
```

```

        "conferenceQueueSlots": 0,
        "firstName": "string",
        "id": "string",
        "isAvailableForConference": true,
        "isAvailableForTransfer": true,
        "lastName": "string",
        "transferQueueSlots": 0
    }
],
"availableSlots": 0,
"conferenceQueueSlots": 0,
"isAvailableForConference": true,
"isAvailableForTransfer": true,
"name": "string",
"transferQueueSlots": 0,
"value": "string"
}
],
"agentGroups": [
    {
        "agentAttributes": [
            {
                "agents": [
                    {
                        "availableSlots": 0,
                        "conferenceQueueSlots": 0,
                        "firstName": "string",
                        "id": "string",
                        "isAvailableForConference": true,
                        "isAvailableForTransfer": true,
                        "lastName": "string",
                        "transferQueueSlots": 0
                    }
                ],
                "availableSlots": 0,
                "conferenceQueueSlots": 0,
                "isAvailableForConference": true,
                "isAvailableForTransfer": true,
                "name": "string",

```

```
        "transferQueueSlots": 0,  
        "value": "string"  
    }  
],  
"agents": [  
  {  
    "availableSlots": 0,  
    "conferenceQueueSlots": 0,  
    "firstName": "string",  
    "id": "string",  
    "isAvailableForConference": true,  
    "isAvailableForTransfer": true,  
    "lastName": "string",  
    "transferQueueSlots": 0  
  }  
],  
"availableAgents": 0,  
"availableEngagementSlots": 0,  
"availableSlots": 0,  
"busyEngagementSlots": 0,  
"canBePrioritize": true,  
"conferenceQueueSlots": 0,  
"disableAgentSelection": true,  
"displayName": "string",  
"engagementsInQueue": 0,  
"id": "string",  
"isAvailableForConference": true,  
"isAvailableForTransfer": true,  
"name": "string",  
"totalAgents": 0,  
"totalEngagementSlots": 0,  
"transferQueueSlots": 0  
}  
],  
"agents": [  
  {  
    "availableSlots": 0,  
    "conferenceQueueSlots": 0,  
    "firstName": "string",
```

```
    "id": "string",
    "isAvailableForConference": true,
    "isAvailableForTransfer": true,
    "lastName": "string",
    "transferQueueSlots": 0
  }
],
"availableAgents": 0,
"availableEngagementSlots": 0,
"availableSlots": 0,
"busyEngagementSlots": 0,
"canBePrioritize": true,
"conferenceQueueSlots": 0,
"disableAgentSelection": true,
"displayName": "string",
"engagementsInQueue": 0,
"id": "string",
"isAvailableForConference": true,
"isAvailableForTransfer": true,
"name": "string",
"siteId": "string",
"totalAgents": 0,
"totalEngagementSlots": 0,
"transferQueueSlots": 0
}
]
```



# transferConferenceOptions (GET)

Retrieves the available options for transfer and conference for the specified engagement.

URI	HTTP Method	Formats
transferConferenceOptions	GET	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent.	string	query	yes
engagementId	Identification number of the engagement.	string	query	yes
registeredId	Id retrieved after agent login	string	query	yes

### Example Request

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/transferConferenceOptions?agentId=123&registeredId=123&engagementId=123&output=json>

**Response Class**

Status 200

```

{
  "businessUnit": [
    {
      "agentAttributes": [
        {
          "agents": [
            {
              "availableSlots": 0,
              "conferenceQueueSlots": 0,
              "firstName": "string",
              "id": "string",
              "isAvailableForConference": true,
              "isAvailableForTransfer": true,
              "lastName": "string",
              "transferQueueSlots": 0
            }
          ],
          "availableSlots": 0,
          "conferenceQueueSlots": 0,
          "isAvailableForConference": true,
          "isAvailableForTransfer": true,
          "name": "string",
          "transferQueueSlots": 0,
          "value": "string"
        }
      ],
      "agentGroups": [
        {
          "agentAttributes": [
            {
              "agents": [
                {
                  "availableSlots": 0,
                  "conferenceQueueSlots": 0,
                  "firstName": "string",
                  "id": "string",
                  "isAvailableForConference": true,

```

```
        "isAvailableForTransfer": true,
        "lastName": "string",
        "transferQueueSlots": 0
      }
    ],
    "availableSlots": 0,
    "conferenceQueueSlots": 0,
    "isAvailableForConference": true,
    "isAvailableForTransfer": true,
    "name": "string",
    "transferQueueSlots": 0,
    "value": "string"
  }
],
"agents": [
  {
    "availableSlots": 0,
    "conferenceQueueSlots": 0,
    "firstName": "string",
    "id": "string",
    "isAvailableForConference": true,
    "isAvailableForTransfer": true,
    "lastName": "string",
    "transferQueueSlots": 0
  }
],
"availableAgents": 0,
"availableEngagementSlots": 0,
"availableSlots": 0,
"busyEngagementSlots": 0,
"canBePrioritize": true,
"conferenceQueueSlots": 0,
"disableAgentSelection": true,
"displayName": "string",
"engagementsInQueue": 0,
"id": "string",
"isAvailableForConference": true,
"isAvailableForTransfer": true,
"name": "string",
```

```
        "totalAgents": 0,  
        "totalEngagementSlots": 0,  
        "transferQueueSlots": 0  
    }  
],  
"agents": [  
    {  
        "availableSlots": 0,  
        "conferenceQueueSlots": 0,  
        "firstName": "string",  
        "id": "string",  
        "isAvailableForConference": true,  
        "isAvailableForTransfer": true,  
        "lastName": "string",  
        "transferQueueSlots": 0  
    }  
],  
"availableAgents": 0,  
"availableEngagementSlots": 0,  
"availableSlots": 0,  
"busyEngagementSlots": 0,  
"canBePrioritize": true,  
"conferenceQueueSlots": 0,  
"disableAgentSelection": true,  
"displayName": "string",  
"engagementsInQueue": 0,  
"id": "string",  
"isAvailableForConference": true,  
"isAvailableForTransfer": true,  
"name": "string",  
"siteId": "string",  
"totalAgents": 0,  
"totalEngagementSlots": 0,  
"transferQueueSlots": 0  
}  
]  
}
```

# transferEngagement (POST)

This REST call is used to post a transfer request only to the server.

URI	HTTP Method	Formats
transferEngagement	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentAttributes	An Array ["name", "value"] (Required only when agent depends on agent attribute for transfer.)	array	body	no
<ul style="list-style-type: none"> <li>name</li> </ul>	Value of agentAttributes array (attribute name)	string	body	no
<ul style="list-style-type: none"> <li>value</li> </ul>	Value of agentAttributes array (attribute value)	string	body	no
agentGroupld	Identification number of the agent group.	string	body	no
agentGroupName	Name of the agentgroup being used	string	body	no
agentld	Unique ID of the agent.	string	body	yes
agentNotes	Notes that agent wants to include	string	body	no
businessUnitld	Identification number of the business unit.	string	body	no
businessUnitName	Name of the business unit	string	body	no
engagementld	Identification number of the engagement.	string	body	yes
prioritizedRequest	true   false	boolean	body	no

Name	Value	Data Type	Parameter Type	Required
targetAgentId	Login name for the target agent. (Required only in cases where a specific agent is picked directly for transfer.)	string	body	no
request-agent-attribute				
• name	Value of request-agent-attributes array (attribute name)	string	body	no
• value	Value of request-agent-attributes array (attribute value)	string	body	no
registeredId	Id retrieved after agent login	string	query	yes

### Example Request

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/transferEngagement?registeredId=123&output=json>

```
{
  "agentAttributes": [
    {
      "name": "string",
      "value": "string"
    }
  ],
  "agentGroupId": "string",
  "agentGroupName": "string",
  "agentId": "string",
  "agentNotes": "string",
  "businessUnitId": "string",
  "businessUnitName": "string",
  "engagementId": "string",
  "prioritizeRequest": true,
  "targetAgentId": "string"
}
```

**Response Class**

Status 200

```
{  
  "engagementId": "string",  
  "reason": "string",  
  "status": "string"  
}
```

# transferOwnership (POST)

Makes a request to transfer ownership of an engagement to another agent.

URI	HTTP Method	Formats
transferOwnership	POST	XML or JSON

## Parameters

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID for the agent transferring the engagement.	string	body	yes
engagementId	Identification number of the engagement to be transferred.	string	body	yes
targetAgentId	Login name for the agent to whom ownership of the engagement will be transferred. .	string	body	no
registeredId	Id retrieved after agent login	string	query	yes

### Example Request URL

URL:

`https://api.touchcommerce.com/engagementAPI/v2/agent/transferOwnership?registeredId=123&output=json`

```
{
  "agentId": "string",
  "engagementId": "string",
  "targetAgentId": "string"
}
```



**Response Class**

Status 200

```
{  
  "engagementId": "string",  
  "reason": "string",  
  "status": "string"  
}
```

# verifySession (POST)

This endpoint verifies that the agent still has the correct session in the system. If the agent gets a 401 error code, he needs to send a /login request again to create a new session.

URI	HTTP Method	Formats
verifySession	POST	XML or JSON

## Parameter

Name	Value	Data Type	Parameter Type	Required
agentId	Unique ID of the agent.	string	body	yes
registeredId	Id retrieved after agent login	string	query	yes

## Example Request

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/verifySession?registeredId=123&output=json>

```
{
  "agentId": "string"
}
```

## Response Class

Status 200

# webrtc/action (POST)

This REST call is used to send webRTC status messages.

URI	HTTP Method	Formats
webrtc	POST	XML or JSON

## Parameter

Name	Value	Data Type	Parameter Type	Required
action	The specified webrtc action such as start/stop call. For example: start/stop call = ['DIALED', 'NETWORK_ERR', 'END_WEBCALL']stringEnum:"DIALED", "NETWORK_ERR", "END_WEBCALL"	string	body	yes
agentId	Unique ID of the agent.	string	body	yes
engagementId	Id retrieved after agent login	string	query	yes

## Example Request

URL:

<https://api.touchcommerce.com/engagementAPI/v2/agent/webrtc/action?registeredId=123&output=json>

```
{
  "action": "DIALED",
  "agentId": "string",
  "engagementId": "string"
}
```

**Response Class**  
Status 200