



## Chat API Reference Guide

Valid for CampusNexus CRM 13.1.X or higher

### Anthology Inc.

5201 Congress Avenue  
Boca Raton, FL 33431  
Main: +1.561.923.2500  
Support: +1.800.483.9106  
[www.anthology.com](http://www.anthology.com)

© 2021 Anthology Inc. All rights reserved.

Anthology, the Anthology logo, Talisma, and CampusNexus are exclusive trademarks of Anthology Inc.

Microsoft, SQL Server, Windows, Windows Server, Internet Explorer, Outlook, Excel, and Hyper-V are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Adobe and Captivate are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Information is subject to change.

#### **CONFIDENTIALITY NOTICE:**

The information contained in this document is confidential. It is the property of Anthology Inc. and shall not be used, disclosed or reproduced without the express written consent of Anthology Inc.

## Contents

Overview .....	4
APIs .....	4
1. Starting a Chat Session.....	4
API Name: Start.....	4
Example.....	4
Response From the Chat Server.....	4
2. Adding a Snippet to a Chat Session.....	5
API Name: AddSnippet.....	5
Example.....	5
Response From the Chat Server.....	5
3. Visitor Side Polling Of a Chat Session.....	6
API Name: Poll.....	6
Example.....	6
Response From the Chat Server.....	6
4. Typing Indicator in a Chat Session .....	7
API Name: TypingIndicator .....	7
Example.....	7
Response From the Chat Server.....	7
5. Uploading a File to a Chat Session .....	7
API Name: UploadFile .....	7
Example.....	7
Response From the Chat Server.....	8
6. Visitor Closes the Browser or Suspends a Chat Session .....	8
API Name: Suspend.....	8
Example.....	8
Response From the Chat Server.....	8
7. Recovering a Chat Session .....	8
API Name: Recover .....	8
Example.....	8
Response From the Chat Server.....	9
8. Ending a Chat Session .....	10
API Name: End .....	10
Example.....	10
Response From the Chat Server.....	10

9. Visitor Submits Feedback On a Chat Session .....	10
API Name: Feedback .....	10
Example.....	10
Response From the Chat Server.....	11

## Overview

This document explains APIs that can be used to perform different tasks in the CampusNexus CRM Chat Server.

Using this document, a Client application written in any platform can communicate with CampusNexus CRM Chat as a visitor.

All methods in this document:

- Can work with both http and https protocols.
- Must include the value of the **Api Key** attribute that's generated in the **Web.config** file of the Media Web component when it's installed.

## APIs

### 1. Starting a Chat Session

API Name: Start

This API initiates a chat a session from any social media platform.

Example

**Start URI:** http://<ChatServerName>/Media/api/Chat/Start

**Request Body:**

```
{  
  "name": "<Value>",  
  "email": "<Value>",  
  "CustomProperties": [{"Id": "<Value>", "Name": "<Value>"}],  
  "PropertiesForContactResolution": [{"Id": "<Value>", "Name": "<Value>"}]}
```

Name	Description	Mandatory
Name	Name of the chat visitor.	No
Email	The visitor's email address.	No
CustomProperties	Custom message object properties that can be used to obtain extra information from the chat visitor. <ul style="list-style-type: none"><li>• <b>ID:</b> The ID of the custom property created for the Message object.</li><li>• <b>Name:</b> Value of the property</li></ul>	No
PropertiesForContactResolution	The Id will be contact properties that resolve the contact, and its corresponding value will be passed in the Name field.	No

Response From the Chat Server

Example:

```
{  
  "RequestId": value,  
  "Guid": "{value}",  
  "PositionInQueue": value,  
  "WaitTime": value,  
  "ReturnCode": value,  
  "Error": "value"  
}
```

Name	Description
RequestId	New request id which gets generated.
Guid	Unique GUID that gets generated for each new request.
PositionInQueue	The current position in the chat queue.
WaitTime	The visitor's wait time.
ReturnCode	The success / failure code.
Error	The error message if there is a failure.

### Contact Resolution in Reactive and Proactive Chat

In reactive chat, when a visitor initiates a chat request, a duplicate check is performed to verify whether the visitor is a contact. If so, the visitor is resolved to a matching contact. If the visitor is not a contact, a new contact is created when the CampusNexus CRM user ends the chat session. In proactive chat, when a visitor accepts a chat request, a duplicate check is performed to verify whether the visitor is already a contact. Contact resolution is performed in the following order:

- i. By specifying custom duplicate check logic for the contact object based on customer requirements. To enable this functionality, the `sproc_CustomDuplicateCheck_ForChat` stored procedure is provided. For more information about this stored procedure, contact Product Support.
- ii. By using duplicate check configurations defined for the contact object in the Data Management Utility.
- iii. By using the email ID of the visitor.

## 2. Adding a Snippet to a Chat Session

API Name: AddSnippet

This API adds the visitor's snippet messages to a chat session.

Example

AddSnippet URI: `http://<chat server>/Media/api/Chat/AddSnippet`

**Request Body:**

```
{
  "requestId":364,
  "snippet":"Can you send me the Event location?"}
```

Name	Description	Mandatory
requestId	The visitors chat session ID	Yes
Snippet	The added snippet.	The visitor's chat snippet text.

Response From the Chat Server

```
{
  "ReturnCode": 0,
  "Error": ""
}
```

Name	Description
ReturnCode	The success / failure code.
Error	The error message if there is a failure.

### 3. Visitor Side Polling Of a Chat Session

API Name: Poll

This API fetches and renders snippet data in the chat session.

Example

**Poll URI:** http://<Chat server>/Media/api/Chat/Poll

**Request Body:**

```
{ "requestId":value,
  "rowId":"value" }
```

Name	Description	Mandatory
requestId	The ID of the chat request.	Yes
Rowid	The row ID from which the visitor needs chat data.	Yes

Response From the Chat Server

**Example:**

```
{
  "Rowid": 8,
  "Agentstatus": "Active",
  "Guid": "",
  "Snippets": [
    {
      "Type": 5,
      "Owner": "",
      "Time": "2020/02/26 17:29:40",
      "SnippetData": "Welcome to Talisma chat.",
      "Options": ""
    },
    {
      "Type": 5,
      "Owner": "",
      "Time": "2020/02/26 17:29:45",
      "SnippetData": "The session has been accepted.",
      "Options": ""
    },
    {
      "Type": 2,
      "Owner": "{sam1}",
      "Time": "2020/02/26 17:30:19",
      "SnippetData": "hello.. how may I help You?",
      "Options": ""
    },
    {
      "Type": 1,
      "Owner": "{Visitor}",
      "Time": "2020/02/26 17:31:15",
      "SnippetData": "Can you send me the Event location?",
      "Options": ""
    }
  ],
  "ReturnCode": 0,
  "Error": ""
}
```

Name	Description
Rowid	The row number that the visitor will expect data from the agent or vice versa. For example, if 3 rows of data have been typed, the value that will be returned by this parameter will now be 4.
Agentstatus	Indicates whether the agent is active, not active or typing.
Guid	Unique GUID which gets generated for each new chat request.
Snippets	Has the following values: <ul style="list-style-type: none"> <li>• <b>Type:</b> Indicates whether the message is generated by the system(5), the agent(2) or the visitor(1).</li> <li>• <b>Owner:</b> Indicates the owner of the snippet.,</li> <li>• <b>Time:</b> Time stamp of the snippet.</li> <li>• <b>SnippetData:</b> The snippet.</li> </ul>
ReturnCode	The success / failure code.
Error	The error message if there is a failure.

#### 4. Typing Indicator in a Chat Session

API Name: TypingIndicator

This API indicates to the agent that the visitor is typing a response.

Example

**TypingIndicator URI:** http://<chat server>/Media/api/Chat/TypingIndicator

**Request Body:**

```
{
  "requestId":value,
  "Action":value}
```

Name	Description	Mandatory
RequestId	The chat request ID.	Yes
Action	Can have one of the following values: <ul style="list-style-type: none"> <li>• 1 – Indicates that the visitor is typing.</li> <li>• 0 – Indicates that the visitor is not typing.</li> </ul>	Yes

Response From the Chat Server

```
{
  "ReturnCode": 0,
  "Error": ""
}
```

Name	Description
ReturnCode	The success / failure code.
Error	Error description in the event of a failure.

#### 5. Uploading a File to a Chat Session

API Name: UploadFile

This API uploads a visitor's file in a chat session.

Example

**UploadFile URI:** http://<chat server>/Media/api/Chat/UploadFile

### Requests sent as parameters:

1. RequestId
2. GUID
3. FileUpload

Name	Description	Mandatory
RequestId	The visitor's request to upload a file in the chat session.	Yes
GUID	The unique GUID that's generated.	Yes
FileUpload	The uploaded file.	Yes

### Response From the Chat Server

```
{  
  "ReturnCode": 0,  
  "Error": ""  
}
```

Name	Description
ReturnCode	The success / failure code.
Error	The error message if there is a failure.

## 6. Visitor Closes the Browser or Suspends a Chat Session

API Name: Suspend

**Suspend URI:** http://<chat server>/Media/api/Chat/Suspend?requestId=<RequestId sent as a Parameter >

Example

Name	Description	Mandatory
requestId	The chat session that the visitor wants to suspend.	Yes

### Response From the Chat Server

```
{  
  "ReturnCode": 0,  
  "Error": ""  
}
```

Name	Description
ReturnCode	The success / failure code.
Error	Error description in the event of a failure.

## 7. Recovering a Chat Session

API Name: Recover

This API recovers a suspended chat session.

Example

**Recover URI:** http://<Chat server>/Media/api/Chat/Recover?requestId=<RequestId sent as a parameter>

Name	Description	Mandatory
requestId	Request that the visitor wants to re-join the chat session.	Yes

## Response From the Chat Server

Example:

```
{
  "Rowid": 8,
  "Agentstatus": "Active",
  "Guid": "",
  "Snippets": [
    {
      "Type": 5,
      "Owner": "",
      "Time": "2020/02/26 17:29:40",
      "SnippetData": "Welcome to Talisma chat.",
      "Options": ""
    },
    {
      "Type": 5,
      "Owner": "",
      "Time": "2020/02/26 17:29:45",
      "SnippetData": "The session has been accepted.",
      "Options": ""
    },
    {
      "Type": 2,
      "Owner": "{sam1}",
      "Time": "2020/02/26 17:30:19",
      "SnippetData": "hello.. how may I help You?",
      "Options": ""
    },
    {
      "Type": 1,
      "Owner": "{111}",
      "Time": "2020/02/26 17:31:15",
      "SnippetData": "Can you send me the Event location?",
      "Options": ""
    }
  ],
  "ReturnCode": 0,
  "Error": ""
}
```

Name	Description
Rowid	The ID of the most recent typed row – from the agent or the visitor.
Agentstatus	Indicates whether the agent is active or not.
Guid	Unique GUID which gets generated for each new chat request.
Snippets	Has the following values: <ul style="list-style-type: none"><li>• <b>Type</b>: Indicates whether the message is generated by the system, the agent or the visitor.</li><li>• <b>Owner</b>: Indicates the owner of the snippet.</li><li>• <b>Time</b>: Time stamp of the snippet.</li><li>• <b>SnippetData</b>: The snippet.</li></ul>
ReturnCode	The success / failure code.
Error	The error message if there is a failure.

## 8. Ending a Chat Session

API Name: End

This API ends a chat session.

Example

**End URI:** http://<chat server>/Media/api/Chat/End?requestId=<RequestId sent as a Parameter >

Name	Description	Mandatory
requestid	The chat request that the visitor wants to end.	Yes

Response From the Chat Server

Example:

```
{
  "ReturnCode": 0,
  "Error": ""
}
```

Name	Description
ReturnCode	The success / failure code.
Error	Error description if the action to end the chat session fails. OR Error description in the event of a failure.

## 9. Visitor Submits Feedback On a Chat Session

API Name: Feedback

This API triggers the visitor's feedback on the chat session.

Example

**Feedback URI:** http://<chat server>/Media/api/Chat/Feedback

**Request Body:**

```
{
  "requestId":value,
  "name":"value",
  "email":"email address",
  "CustomProperties":[{"Id":Value,"Name":"Value"}]}
```

Name	Description	Mandatory
requestId	The ID of the Chat session	Yes
Name	The visitor's name.	No
Email	The visitor's email address to which the chat transcript can be sent.	The email is required to use this functionality. Otherwise, this field can be blank.
CustomProperties	Group of custom message object properties which can be used to get feedback from the visitor.	Yes (if required)

## Response From the Chat Server

```
{  
  "ReturnCode": 0,  
  "Error": ""  
}
```

Name	Description
ReturnCode	The success / failure code.
Error	Error description in the event of a failure.