



# Community Resource Operations

A Cisco WebEx Social community is a virtual group of Cisco WebEx Social users. Typically, these users share interests or goals. Users who belong to a community have access to the blogs, wikis, forums, library, and dashboard of the community. There are three types of communities:

- Open—Communities that anyone can join. Content in open communities appears in search results.
- Restricted—Communities for which someone must request and be granted membership to join. Content in restricted communities appears in search results.
- Hidden—Communities that someone can join by invitation only. Hidden communities do not appear on the Communities page, and content in communities does not appear in search results.

The Cisco WebEx Social API Community resource operations let you obtain information about communities and perform various actions with communities.

[Table 5-1](#) provides a summary of the Community resource operations. Each operation is described in detail in the section that is listed.

These sections provide this information:

- Purpose—Brief explanation of the operation
- Availability—Cisco WebEx Social releases that support the operation
- Request—HTTP method and URI for the operation
- Request Payload—Description of the request payload, if the operation requires one
- Response—Description of the expected response when the operation executes
- Example—Sample request and response for the operation

In these sections, “you” and “yourself” refer to the authenticated user who is executing an operation.

**Table 5-1** *Community Resource Operations Summary*

Operation and Reference	Purpose
<a href="#">Get Information about a Community, page 5-4</a>	Retrieves information about the designated community
<a href="#">Get Information about Communities, page 5-6</a>	Retrieves information about communities that are visible to you
<a href="#">Create a Community, page 5-8</a>	Creates a community
<a href="#">Update a Community, page 5-10</a>	Updates the designated community
<a href="#">Delete a Community, page 5-12</a>	Deletes the designated community from Cisco WebEx Social

**Table 5-1 Community Resource Operations Summary (continued)**

<b>Operation and Reference</b>	<b>Purpose</b>
Add Yourself to a Community, page 5-13	Adds you to the designated community
Remove Yourself from a Community, page 5-15	Removes you from the designated community
Add a User to a Community, page 5-16	Adds the designated user to the designated community
Add a User to a Community (Alternate Method), page 5-18	Adds the designated user to the designated community
Remove a User from a Community, page 5-20	Removes the designated user from the designated community
Remove a User from a Community (Alternate Method), page 5-21	Removes the designated user from the designated community
Assign the Community Owner Role to a User, page 5-22	Assigns the Community Owner role to the designated user for the designated community
Remove the Community Owner Role from a User, page 5-23	Removes the Community Owner role from the designated user for the designated community
Get Information about Users Assigned to a Community, page 5-24	Retrieves information about users that are assigned to the designated community
Get Information about Users Assigned to a Community (Alternate Method), page 5-26	Retrieves information about users that are assigned to the designated community
Add a User Group to a Community, page 5-28	Adds the designated user group to the designated community
Remove a User Group from a Community, page 5-29	Removes the designated user group from the designated community
Assign the Community Owner Role to Users in a User Group, page 5-30	Assigns the Community Owner role to the users in the designated user for the designated community
Remove the Community Owner Role from Users in a User Group, page 5-31	Removes the Community Owner role from the users in the designated user for the designated community
Get Information about User Groups Assigned to a Community, page 5-32	Retrieves information about user groups that are assigned to the designated community
Get Information about a Community Category, page 5-34	Retrieves information about the designated community category
Get Information about Community Categories, page 5-36	Retrieves information about community categories
Get Community Types that a Community Category Supports, page 5-38	Retrieves the community types that the designated community category supports
Get Information about Membership Requests for a Community, page 5-40	Retrieves information about membership requests that have been made to the designated community
Get Information about Posts for a Community, page 5-42	Retrieves information about all posts that have been shared with the designated community
Get Information about Questions and Answers for a Community, page 5-44	Retrieves information about questions and answers that were submitted when the designated community was created

**Table 5-1 Community Resource Operations Summary (continued)**

<b>Operation and Reference</b>	<b>Purpose</b>
<a href="#">Get Information about the Document Library of a Community, page 5-46</a>	Retrieves information about the document library of a designated community
<a href="#">Get Information about Document Files in the Document Library of a Community, page 5-48</a>	Retrieves information about all document files in that are in the document library of the designated community
<a href="#">Get Information about Document Files in the Document Library of a Community that are Authored by a Designated User, page 5-50</a>	Retrieves information about document files that are in the document library of the designated community and that are authored by the designated user
<a href="#">Get Information about All Custom Attributes of a Community, page 5-52</a>	Retrieves information about all custom attributes of the designated community
<a href="#">Update Custom Attributes of a Community, page 5-54</a>	Updates designated custom attributes for the designated community
<a href="#">Get Information about Community Profile Images, page 5-56</a>	Retrieves information about profile images for the designated community
<a href="#">Update a Community Profile Image, page 5-58</a>	Updates the community profile image for the designated community
<a href="#">Get All Activities that are Visible to You in a Community, page 5-59</a>	Retrieves information about all activities that are visible to you in the designated community
<a href="#">Get Community Metrics, page 5-61</a>	Retrieves metrics about various activities that occurred in a community.
<a href="#">Get Code of Conduct, page 5-63</a>	Retrieves the Cisco WebEx Social community code of conduct

# Get Information about a Community

**Purpose** Retrieves information about the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
GET	{ <a href="#">http https</a> }://server[:port]/api/quad/rest/communities/community_id[?query_parameters]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve information. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the Community resource to include in a response ([Table 3-9 on page 3-11](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter does not apply
  - The **startIndex** query parameter does not apply
  - The **count** query parameter does not apply
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one Community resource (the [“Community Resource” section on page 3-10](#) describes this resource).

**Example**

**Request**

```
GET http://webexsocialserver/api/quad/rest/communities/1234
Accept:application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
```

The response can also include the Community resource. For an example, see the [“Community Resource Response: JSON Format, One Resource Returned”](#) section on page 18-47

# Get Information about Communities

**Purpose** Retrieves information about communities that are visible to you.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
GET	{ <a href="#">http</a> { <a href="#">https</a> }://server[:port]/api/quad/rest/communities[?query_parameters]}

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *query\_parameters*—Query parameters. See the “[Query Parameters](#)” section on page 1-20 for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the Community resource to include in a response ([Table 3-9 on page 3-11](#) describes these fields)
  - The **filterBy** query parameter is supported for filtering resources in a response by the **type** field in the Community resource
  - The **sortOrder** query parameter is supported for sorting resources in a response by the **name**, **stateChangedDate**, or **type** field in the Community resource
  - The **startIndex** query parameter is supported for paginating return results
  - The **count** query parameter is supported for designating the maximum number of items in the return results
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response (see the “[API Responses](#)” section on page 1-17) with a response payload that can include one or more Community resources (the “[Community Resource](#)” section on page 3-10 describes this resource).

**Example**

**Request**

```
GET http://webexsocialserver/api/quad/rest/communities
Accept: application/json
```

**Response**

```
HTTP/1.1 200 OK  
Content-Type: application/json
```

The response can also include the Community resource. For an example, see the [“Community Resource Response: JSON Format, Multiple Resources Returned”](#) section on page 18-49.

# Create a Community

**Purpose** Creates a community.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
POST	{http https}://server[:port]/api/quad/rest/communities[?query_parameters]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *query\_parameters*—Query parameters. See the “[Query Parameters](#)” section on page 1-20 for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the Community resource to include in a response ([Table 3-9 on page 3-11](#) describes these fields).
  - The **filterBy** query parameter is not supported for filtering resources in a response.
  - The **sortOrder** query parameter does not apply.
  - The **startIndex** query parameter does not apply.
  - The **count** query parameter does not apply.
  - The **meta** query parameter does not apply.
  - The special query parameter **agreeToCodeOfConduct={true|false}** is required if Cisco WebEx Social is configured so that a user must agree to the Cisco WebEx Social community code of conduct when creating a community (see the “[CodeOfConduct Resource](#)” section on [page 3-10](#) for related information). In this case, to create a community, you must set this query parameter to **true** to indicate that you agree to the code of conduct.



**Request Payload**

The request payload includes fields from the Community resource ([Table 3-9 on page 3-11](#) describes these fields):

Required Fields	Optional Fields
description name tags type	communityCategory <sup>1</sup> key questionAnswer <sup>2</sup> value

1. The **id** field in the communityCategory resource is optional. The system ignores other fields in this resource.
2. The **questionId** and the **answer** fields in the questionAnswer resource are optional. The system ignores other fields in this resource.

**Response**

Standard response (see the “[API Responses](#)” section on page 1-17) with a response payload that can include one Community resource (the “[Community Resource](#)” section on page 3-10 describes this resource).

**Example****Request**

```
POST http://webexsocialserver/api/quad/rest/communities
Accept: application/json
Content-Type: application/json
```

```
{
  "category": "general",
  "description": "To capture Project Video activities",
  "name": "Project Video",
  "type": "open",
  "tags": [
    {
      "resource": "http://quad.cisco.com/schema/1.0/tag",
      "name": "Cisco WebEx Social"
    }
  ]
}
```

**Response**

```
HTTP/1.1 201 Created
Location: http://webexsocialserver/api/quad/rest/communities/1234
Content-Type: application/json
```

The response can also include the Community resource. For an example, see the “[Community Resource Response: JSON Format, One Resource Returned](#)” section on page 18-47.

# Update a Community

**Purpose** Updates information about the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
PUT	{ <a href="#">httphttps</a> ://server[:port]/api/quad/rest/communities/community_id [?query_parameters]}

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community to update. For information about obtaining the identifier that you need, see the “[Identifiers of Cisco WebEx Social Entities](#)” section on page 1-12.
- *query\_parameters*—Query parameters. See the “[Query Parameters](#)” section on page 1-20 for information about using query parameters. For this operation:
  - The **fields** query parameter does not apply
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter does not apply
  - The **startIndex** query parameter does not apply
  - The **count** query parameter does not apply
  - The **meta** query parameter does not apply

**Request Payload**

The request payload includes fields from the Community resource ([Table 3-9 on page 3-11](#) describes these fields):

Required Fields	Optional Fields
—	name shortName state <sup>1</sup> tags

1. If you specify the **state** field, the system ignores any other Community resource field that is included in the request payload.

**Response**

Standard response (see the “[API Responses](#)” section on page 1-17) with an HTTP header only.

**Example****Request**

```
PUT http://webexsocialserver/api/quad/rest/communities/4810201
Accept: application/json
Content-Type: application/json
```

```
{
  "description": "To capture Project Video related activities in Sally's team"
}
```

**Response**

```
HTTP/1.1 204 No Content
```

# Delete a Community

**Purpose** Deletes the designated community from Cisco WebEx Social.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
DELETE	{ <a href="#">httphttps</a> }://server[:port]/api/quad/rest/communities/community_id

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community to delete. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

**Example**

**Request**

```
DELETE http://webexsocialserver/api/quad/rest/communities/4810201
```

**Response**

```
HTTP/1.1 204 No Content
```

# Add Yourself to a Community

## Purpose

Adds you to the designated community.

If you have the system administrator privilege in Cisco WebEx Social, this operation adds you to the community and no membership request is generated.

If you do not have the system administrator privilege, this operation adds you to the community only if the community is an Open type. For other types of communities, the system creates a membership request.

## Availability

Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
POST	{ <a href="#">http</a> { <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/users/@me</a> }

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community to which to add yourself. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

## Request Payload

A request payload is not required.

## Response

If no membership request is generated, this operation returns a standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

If a membership request is generated, this operation returns a standard response (see the [“API Responses” section on page 1-17](#)) that includes the MembershipRequest resource (the [“MembershipRequest Resource” section on page 3-30](#) describes this resource).

## Example

### Request

```
POST http://webexsocialserver/api/quad/rest/communities/4810301/users/@me
Accept: application/json
content-type: json
```

### Response—Membership request is not generated

```
HTTP/1.1 204 No Content
```

**Response—Membership request is generated**

HTTP/1.1 303 See Other

Location: http://webexsocialserver/api/quad/rest/membershiprequests/210747

```

{
  "entry": [
    {
      "data": {
        "approver": null,
        "community": {
          "description": "second",
          "id": 4710003,
          "name": "Dos",
          "shortName": "/dos",
          "state": "live",
          "type": "restricted"
        },
        "createDate": 1319730564972,
        "id": 210747,
        "replyComments": "",
        "replyDate": 0,
        "requestComments": "",
        "requester": {
          "firstName": "Pat",
          "fullName": "Pat Kim",
          "id": 1234,
          "images": [ ],
          "jobTitle": "",
          "lastName": "Kim",
          "screenName": "patkim"
        },
        "resource": "http://quad.cisco.com/schema/1.0/membershiprequest",
        "status": "pending"
      }
    }
  ],
  "filtered": false,
  "itemsPerPage": 1,
  "serverMessages": null,
  "sorted": false,
  "startIndex": 0,
  "statusCode": 200,
  "totalResults": 1
}

```

# Remove Yourself from a Community

**Purpose** Removes you from the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

Request	HTTP Method	URI
	DELETE	{ <a href="#">http</a> { <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/users/@me</a>

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community from which to remove yourself. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

## Example

### Request

```
DELETE http://webexsocialserver/api/quad/rest/communities/4810301/users/@me
```

### Response

```
HTTP/1.1 204 No Content
```

# Add a User to a Community

## Purpose

Adds the designated user to the designated community. The user is assigned the Community Member role automatically.

If the designated user is a user other than yourself, you must be a Cisco WebEx Social administrator or an owner or administrator the community to add that user to the community. In this case, no membership request is generated.

If you are not a Cisco WebEx Social administrator or an owner or administrator of the community, you can use this operation to add yourself to the community only if the community is an Open type. For other types of communities, the system generates a membership request.

## Availability

Cisco WebEx Social 3.3 and later.

This operation is deprecated beginning with Cisco WebEx Social 3.0 SR2. From that release on, use the operation that the [“Add a User to a Community \(Alternate Method\)”](#) section on page 5-18 describes.

## Request

HTTP Method	URI
POST	<code>{http https}://server[:port]/api/quad/rest/communities/community_id/users/user_id</code>

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community to which to add the user. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities”](#) section on page 1-12.
- *user\_id*—Identifier of the user to add to the community. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities”](#) section on page 1-12.

## Request Payload

A request payload is not required.

## Response

If no membership request is generated, this operation returns a standard response (see the [“API Responses”](#) section on page 1-17) with an HTTP header only.

If a membership request is generated, this operation returns a standard response (see the [“API Responses”](#) section on page 1-17) that includes the MembershipRequest resource (the [“MembershipRequest Resource”](#) section on page 3-30 describes this resource).



**Example****Request**

```
POST http://webexsocialserver/api/quad/rest/communities/4710003/users/4510003
Accept: application/json
Content-Type: application/json
```

**Response—Membership request is not generated**

```
HTTP/1.1 204 No Content
```

**Response—Membership request is generated**

```
HTTP/1.1 303 See Other
Location: http://webexsocialserver/api/quad/rest/membershiprequests/210747
{
  "entry": [
    {
      "data": {
        "approver": null,
        "community": {
          "description": "second",
          "id": 4710003,
          "name": "Dos",
          "shortName": "/dos",
          "state": "live",
          "type": "restricted"
        },
        "createDate": 1319730564972,
        "id": 210747,
        "replyComments": "",
        "replyDate": 0,
        "requestComments": "",
        "requester": {
          "firstName": "Pat",
          "fullName": "Pat Kim",
          "id": 1234,
          "images": [ ],
          "jobTitle": "",
          "lastName": "Kim",
          "screenName": "patkim"
        },
        "resource": "http://quad.cisco.com/schema/1.0/membershiprequest",
        "status": "pending"
      }
    }
  ],
  "filtered": false,
  "itemsPerPage": 1,
  "serverMessages": null,
  "sorted": false,
  "startIndex": 0,
  "statusCode": 200,
  "totalResults": 1
}
```

# Add a User to a Community (Alternate Method)

## Purpose

Adds the designated user to the designated community.

If the designated user is a user other than yourself, you must be a Cisco WebEx Social administrator or an owner or administrator of the community to add that user to the community. In this case, no membership request is generated.

If you are not a Cisco WebEx Social administrator or an owner or administrator of the community, you can use this operation to add yourself to the community only if the community is an Open type. For other types of communities, the system generates a membership request.

## Availability

Cisco WebEx Social 3.0 SR 2 and later.

## Request

HTTP Method	URI
POST	{ <a href="#">http https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/communityusers/user_id</a>

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community to which to add the user. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *user\_id*—Identifier of the user to add to the community. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

## Request Payload

A request payload is not required.

## Response

If no membership request is generated, this operation returns a standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

If a membership request is generated, this operation returns a standard response (see the [“API Responses” section on page 1-17](#)) that includes the MembershipRequest resource (the [“MembershipRequest Resource” section on page 3-30](#) describes this resource).

## Example

### Request

```
POST http://webexsocialserver/api/quad/rest/communities/4710003/communityusers/4510003
Accept: application/json
Content-Type: application/json
```

**Response—Membership request is not generated**

HTTP/1.1 204 No Content

**Response—Membership request is generated**

HTTP/1.1 303 See Other

Location: <http://webexsocialserver/api/quad/rest/membershiprequests/210747>

```
{
  "entry": [
    {
      "data": {
        "approver": null,
        "community": {
          "description": "second",
          "id": 4710003,
          "name": "Dos",
          "shortName": "/dos",
          "state": "live",
          "type": "restricted"
        },
        "createDate": 1319730564972,
        "id": 210747,
        "replyComments": "",
        "replyDate": 0,
        "requestComments": "",
        "requester": {
          "firstName": "Pat",
          "fullName": "Pat Kim",
          "id": 1234,
          "images": [ ],
          "jobTitle": "",
          "lastName": "Kim",
          "screenName": "patkim"
        },
        "resource": "http://quad.cisco.com/schema/1.0/membershiprequest",
        "status": "pending"
      }
    }
  ],
  "filtered": false,
  "itemsPerPage": 1,
  "serverMessages": null,
  "sorted": false,
  "startIndex": 0,
  "statusCode": 200,
  "totalResults": 1
}
```

# Remove a User from a Community

## Purpose

Removes the designated user from the designated community.

If the designated user is a user other than yourself, you must be a Cisco WebEx Social administrator or an owner or administrator of the community to remove that user from the community.

If you are not a Cisco WebEx Social administrator or an owner or administrator of the community, you can use this operation to remove yourself from the community.

## Availability

Cisco WebEx Social 3.3 and later.

This operation is deprecated beginning with Cisco WebEx Social 3.0 SR2. From that release on, use the operation that the [“Remove a User from a Community \(Alternate Method\)”](#) section on page 5-21 describes.

## Request

HTTP Method	URI
DELETE	<code>{http https}://server[:port]/api/quad/rest/communities/community_id/users/user_id</code>

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community from which to remove the user. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities”](#) section on page 1-12.
- *user\_id*—Identifier of the user to remove from the community. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities”](#) section on page 1-12.

## Request Payload

A request payload is not required.

## Response

Standard response (see the [“API Responses”](#) section on page 1-17) with an HTTP header only.

## Example

### Request

```
DELETE http://webexsocialserver/api/quad/rest/communities/4710003/users/4510003
```

### Response

```
HTTP/1.1 204 No Content
```

# Remove a User from a Community (Alternate Method)

## Purpose

Removes the designated user from the designated community.

If the designated user is a user other than yourself, you must be a Cisco WebEx Social administrator or an owner or administrator of the community to remove that user from the community.

If you are not a Cisco WebEx Social administrator or an owner or administrator of the community, you can use this operation to remove yourself from the community.

## Availability

Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
DELETE	{ <a href="#">http https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/communityusers/user_id</a>

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community from which to remove the user. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *user\_id*—Identifier of the user to remove from the community. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

## Request Payload

A request payload is not required.

## Response

Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

## Example

### Request

```
DELETE http://webexsocialserver/api/quad/rest/communities/4710003/communityusers/4510003
```

### Response

```
HTTP/1.1 204 No Content
```

# Assign the Community Owner Role to a User

## Purpose

Assigns the Community Owner role to the designated user for the designated community. The user must already be a member of the community.

You must be a Cisco WebEx administrator or an owner or administrator of the designated community to execute this operation successfully.

## Availability

Cisco WebEx Social 3.0 SR2 and later.

## Request

HTTP Method	URI
POST	{http https}://server[:port]/api/quad/rest/communities/community_id/communityusers/user_id/roles/@owner

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community in which the user is to have the Community Owner role. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *user\_id*—Identifier of the user to whom the Community Owner role is to be assigned. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

## Request Payload

A request payload is not required.

## Response

Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

## Example

### Request

```
POST http://quadserver/api/quad/rest/communities/9876/communityusers/1234/roles/@owner
```

### Response

```
HTTP/1.1 204 No Content
```

# Remove the Community Owner Role from a User

## Purpose

Removes the Community Owner role from the designated user for the designated community. This operation does not remove the user from the community.

You must be a Cisco WebEx administrator or an owner or administrator of the designated community to execute this operation successfully.

## Availability

Cisco WebEx Social 3.0 SR2 and later.

## Request

HTTP Method	URI
DELETE	{ <a href="#">http</a> / <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/communityusers/user_id/roles/@owner</a>

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community in which the user is to have the Community Owner role removed. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *user\_id*—Identifier of the user to whom the Community Owner role is to be removed. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

## Request Payload

A request payload is not required.

## Response

Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

## Example

### Request

```
DELETE http://quadserver/api/quad/rest/communities/9876/communityusers/1234/roles/@owner
```

### Response

```
HTTP/1.1 204 No Content
```

# Get Information about Users Assigned to a Community

**Purpose** Retrieves information about users that are assigned to the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

This operation is deprecated beginning with Cisco WebEx Social 3.0 SR2. From that release on, use the operation that the [“Get Information about Users Assigned to a Community \(Alternate Method\)”](#) section on page 5-26 describes.

**Request**

HTTP Method	URI
GET	{ <a href="#">http</a> { <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/users</a> [? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve information about assigned users. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities”](#) section on page 1-12.
- *query\_parameters*—Query parameters. See the [“Query Parameters”](#) section on page 1-20 for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the User resource to include in a response ([Table 3-46 on page 3-42](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter is supported for sorting resources in a response by the **firstName**, **lastName**, **jobTitle**, or **screenName** field in the User resource
  - The **startIndex** query parameter is supported for paginating return results
  - The **count** query parameter is supported for designating the maximum number of items in the return results
  - The **meta** query parameter is supported with the **follow** value for requesting relationship metadata

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses”](#) section on page 1-17) with a response payload that can include one or more User resources (the [“User Resource”](#) section on page 3-42 describes this resource).



**Example****Request**

```
GET http://webexsocialserver/api/quad/rest/communities/9876/users
Accept: application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
```

The response can also include the User resource. For an example, see the [“User Resource Response: JSON Format, Multiple Resources Returned”](#) section on page 18-6.

# Get Information about Users Assigned to a Community (Alternate Method)

**Purpose** Retrieves information about users that are assigned to the designated community.

**Availability** Cisco WebEx Social 3.0 SR2 and later.

## Request

HTTP Method	URI
GET	{ <a href="#">http https</a> }://server[:port]/api/quad/rest/communities/community_id/communityusers[/member_type][?query_parameters]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve information about users. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *member\_type*—You can limit the users that this operation returns from the designated community by specifying any of these parameters:
  - **@directmembers**—Causes the operation to retrieve only users who are not members of any user group that is assigned to the community
  - **@memberrole**—Causes the operation to retrieve only users that have the Community Member role in the community
  - **@ownerrole**—Causes the operation to retrieve only users that have the Community Owner role in the community
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the CommunityUser resource to include in a response ([Table 3-13 on page 3-15](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter is supported for sorting resources in a response by the **firstName**, **lastName**, **jobTitle**, or **screenName** field in the CommunityUser resource
  - The **startIndex** query parameter is supported for paginating return results
  - The **count** query parameter is supported for designating the maximum number of items in the return results
  - The **meta** query parameter is supported with the **follow** value for requesting relationship metadata

---

**Request Payload** A request payload is not required.

---

**Response** Standard response (see the [“API Responses”](#) section on page 1-17) with a response payload that can include one or more CommunityUser resources (the [“CommunityUser Resource”](#) section on page 3-15 describes this resource).

---

**Example**

**Request**

```
GET http://webexsocialserver/api/quad/rest/communities/9876/communityusers
Accept: application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
```

The response can also include the CommunityUser resource. For an example, see the [“CommunityUser Resource Response: JSON Format, Multiple Resources Returned”](#) section on page 18-15.

# Add a User Group to a Community

## Purpose

Adds the designated user group to the designated community. Each user in the user group is assigned the Member role in the community automatically.

You must be a Cisco WebEx administrator or an owner or administrator of the designated community to execute this operation successfully.

## Availability

Cisco WebEx Social 3.0 SR2 and later.

## Request

HTTP Method	URI
POST	{ <a href="#">http</a> { <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/communityusergroups/usergroup_id</a>

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community to which to add the user group. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *usergroup\_id*—Identifier of the user group to add to the community. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

## Request Payload

A request payload is not required.

## Response

Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

## Example

### Request

```
POST http://webexsocialserver/api/quad/rest/communities/4710003/communityusergroups/1154
```

### Response

```
HTTP/1.1 204 No Content
```

# Remove a User Group from a Community

## Purpose

Removes the designated user group from the designated community.

You must be a Cisco WebEx administrator or an owner or administrator of the designated community to execute this operation successfully.

## Availability

Cisco WebEx Social 3.0 SR2 and later.

## Request

HTTP Method	URI
DELETE	{ <a href="#">http</a> / <a href="#">https</a> }:// <i>server[:port]</i> / <a href="#">api/quad/rest/communities/community_id/communityusergroups/usergroup_id</a>

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community from which to remove the user group. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *usergroup\_id*—Identifier of the user group to remove from the community. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

## Request Payload

A request payload is not required.

## Response

Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

## Example

### Request

```
DELETE http://webexsocialserver/api/quad/rest/communities/4710003/communityusergroups/1154
```

### Response

```
HTTP/1.1 204 No Content
```

# Assign the Community Owner Role to Users in a User Group

## Purpose

Assigns the Community Owner role to the users in the designated user for the designated community. Each user in the user group must already be a member of the community.

You must be a Cisco WebEx administrator or an owner or administrator of the designated community to execute this operation successfully.

## Availability

Cisco WebEx Social 3.0 SR2 and later.

## Request

HTTP Method	URI
POST	{ <a href="#">http https</a> }:// <i>server</i> [: <i>port</i> ]/api/quad/rest/communities/ <i>community_id</i> /communityusergroups/ <i>usergroup_id</i> /roles/@owner

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community in which users in the user group are to have the Community Owner role. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *usergroup\_id*—Identifier of the user group that includes the users to which the Community Owner role is to be assigned. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

## Request Payload

A request payload is not required.

## Response

Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

## Example

### Request

```
POST http://quadserver/api/quad/rest/communities/9876/communityusergroups/1154/roles/@owner
```

### Response

```
HTTP/1.1 204 No Content
```

# Remove the Community Owner Role from Users in a User Group

## Purpose

Removes the Community Owner role from the users in the designated user for the designated community. This operation does not remove the users from the community.

You must be a Cisco WebEx administrator or an owner or administrator of the designated community to execute this operation successfully.

## Availability

Cisco WebEx Social 3.0 SR2 and later.

## Request

HTTP Method	URI
DELETE	{http https}://server[:port]/api/quad/rest/communities/community_id/ /communityusergroups/usergroup_id/roles/@owner

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community in which the user is to have the Community Owner role removed. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *user\_id*—Identifier of the user to whom the Community Owner role is to be removed. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

## Request Payload

A request payload is not required.

## Response

Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

## Example

### Request

```
DELETE http://quadserver/api/quad/rest/communities/9876/communityusergroups/1154/roles/@owner
```

### Response

```
HTTP/1.1 204 No Content
```

# Get Information about User Groups Assigned to a Community

**Purpose** Retrieves information about user groups that are assigned to the designated community.

**Availability** Cisco WebEx Social 3.0 SR2 and later.

**Request**

HTTP Method	URI
GET	{ <a href="#">http</a> { <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/communityusergroups</a> [ <i>role</i> ][? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve user assigned user groups. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *role*—You can limit the user groups that this operation returns from the designated community by specifying either of these values:
  - **@memberrole**—Causes the operation to retrieve only user who are in user group and who have the Community Member role in the community
  - **@ownerrole**—Causes the operation to retrieve only user who are in the user group and who have the Community Owner role in the community
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the CommunityUserGroup resource to include in a response ([Table 3-14 on page 3-17](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter is supported for sorting resources in a response by the **description** or **name** field in the CommunityUserGroup resource
  - The **startIndex** query parameter is supported for paginating return results
  - The **count** query parameter is supported for designating the maximum number of items in the return results
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.



**Response**

Standard response (see the [“API Responses”](#) section on page 1-17) with a response payload that can include one or more CommunityUserGroup resources (the [“CommunityUserGroup Resource”](#) section on page 3-17 describes this resource).

**Example****Request**

```
GET http://quadserver/api/quad/rest/communities/9876/communityusergroups
Accept: application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "statusCode": 200,
  "startIndex": 0,
  "itemsPerPage": 1,
  "totalResults": 1,
  "filtered": false,
  "sorted": false,
  "entry": [
    {
      "data": {
        "creator": {
          "firstName": "Pat",
          "fullName": "Pat Kim",
          "id": 10261,
          "images": null,
          "jobTitle": "",
          "lastName": "Kim",
          "resource": "http://quad.cisco.com/schema/1.0/user",
          "screenName": "patkim",
          "status": "ACTIVE"
        },
        "description": "Community Owners User Group",
        "id": 1810101,
        "name": "Community Owners",
        "resource": "http://quad.cisco.com/schema/1.0/communityusergroup",
        "roles": [
          {
            "id": 10218,
            "resource": "http://quad.cisco.com/schema/1.0/role",
            "name": "Owner",
            "description": "Role that provides the community owner privilege to users</description.",
            "title": ""
          }
        ]
      }
    }
  ],
  "serverMessages": null
}
```

# Get Information about a Community Category

**Purpose** Retrieves information about the designated community category.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
GET	{ <a href="#">http</a> { <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/categories/category_id</a> [? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *category\_id*—Identifier of the community category for which to retrieve information. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the CommunityCategory resource to include in a response ([Table 3-10 on page 3-13](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter does not apply
  - The **startIndex** query parameter does not apply
  - The **count** query parameter does not apply
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one CommunityCategory resource (the [“CommunityCategory Resource” section on page 3-13](#) describes this resource).

**Example**

**Request**

```
GET http://webexsocialserver/api/quad/rest/communities/categories/210044
Accept: application/json
```

**Response**

```
HTTP/1.1 200 OK  
Content-Type: application/json
```

The response can also include the CommunityCategory resource. For an example, see the [“CommunityCategory Resource Response: JSON Format, One Resource Returned”](#) section on page 18-53.

# Get Information about Community Categories

**Purpose** Retrieves information about community categories.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
GET	{ <a href="#">http</a> { <a href="#">https</a> }://server[:port]/api/quad/rest/communities/categories [?query_parameters]}

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *query\_parameters*—Query parameters. See the “[Query Parameters](#)” section on page 1-20 for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the CommunityCategory resource to include in a response ([Table 3-10 on page 3-13](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter is supported for sorting resources in a response by the **name** field in the CommunityCategory resource
  - The **startIndex** query parameter is supported for paginating return results
  - The **count** query parameter is supported for designating the maximum number of items in the return results
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response (see the “[API Responses](#)” section on page 1-17) with a response payload that can include one or more CommunityCategory resources (the “[CommunityCategory Resource](#)” section on page 3-13 describes this resource).

**Example**

**Request**

```
GET http://webexsocialserver/api/quad/rest/communities/categories
Accept: application/json
```

**Response**

```
HTTP/1.1 200 OK  
Content-Type: application/json
```

The response can also include `CommunityCategory` resources. For an example, see the [“CommunityCategory Resource Response: JSON Format, Multiple Resources Returned”](#) section on page 18-55.

# Get Community Types that a Community Category Supports

**Purpose** Retrieves the community types (open, restricted, or hidden) that the designated community category supports.

**Availability** Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
GET	{ <a href="#">http</a> / <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/categories/category_id/types</a> [? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *category\_id*—Identifier of the community category for which to retrieve supported community types. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities”](#) section on page 1-12.
- *query\_parameters*—Query parameters. See the [“Query Parameters”](#) section on page 1-20 for information about using query parameters. For this operation:
  - The **fields** query parameter is not supported
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter does not apply
  - The **startIndex** query parameter does not apply
  - The **count** query parameter does not apply
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses”](#) section on page 1-17) with a response payload that can include one or more CommunityType resources (the [“CommunityType Resource”](#) section on page 3-15 describes this resource).

## Example

### Request

```
GET http://webexsocialserver/api/quad/rest/communities/categories/210044/types
Accept: application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "statusCode": 200,
  "startIndex": 0,
  "itemsPerPage": 3,
  "totalResults": -1,
  "filtered": false,
  "sorted": false,
  "entry": [
    {
      "data": "open"
    },
    {
      "data": "hidden"
    },
    {
      "data": "restricted"
    }
  ],
  "serverMessages": null
}
```

# Get Information about Membership Requests for a Community

**Purpose** Retrieves information about membership requests that have been made to the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
GET	{http https}://server[:port]/api/quad/rest/communities/community_id/membershiprequests[?query_parameters]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve membership requests. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the MembershipRequest resource to include in a response ([Table 3-31 on page 3-30](#) describes these fields)
  - The **filterBy** query parameter is supported for filtering resources in a response by the **status** field in the MembershipRequest resource
  - The **sortOrder** query parameter does not apply
  - The **startIndex** query parameter does not apply
  - The **count** query parameter does not apply
  - The **count** query parameter does not apply
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one MembershipRequest resource (the [“MembershipRequest Resource” section on page 3-30](#) describes this resource).



**Example****Request**

```
GET http://webexsocialserver/api/quad/rest/communities/4710003/membershiprequests
Content-Type: application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "entry": [
    {
      "data": {
        "approver": null,
        "community": {
          "description": "second",
          "id": 4710003,
          "name": "Dos",
          "shortName": "/dos",
          "state": "live",
          "type": "restricted"
        },
        "createDate": 1319730564972,
        "id": 110499,
        "replyComments": "",
        "replyDate": 0,
        "requestComments": "",
        "requester": {
          "firstName": "Pat",
          "fullName": "Pat Kim",
          "id": 1234,
          "images": [ ],
          "jobTitle": "",
          "lastName": "Kim",
          "screenName": "patkim"
        },
        "resource": "http://quad.cisco.com/schema/1.0/membershiprequest",
        "status": "pending"
      }
    }
  ],
  "filtered": false,
  "itemsPerPage": 1,
  "serverMessages": null,
  "sorted": false,
  "startIndex": 0,
  "statusCode": 200,
  "totalResults": 0
}
```

# Get Information about Posts for a Community

**Purpose** Retrieves information about all posts that have been shared with the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
GET	{ <a href="#">http</a> / <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/posts</a> [? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve posts. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the Post resource to include in a response ([Table 3-36 on page 3-34](#) describes these fields).
  - The **filterBy** query parameter is supported for filtering resources in a response by the **creator** or **state** field in the Community resource. If you filter by the **creator** field, only the resource that represents you is supported as the field value. If you filter by the **state** field, only **DELETED** is supported as the field value.
  - The **sortOrder** query parameter is supported for sorting resources in a response by the **creator**, **lastModifiedDate**, or **title** field in the Post resource.
  - The **startIndex** query parameter is supported for paginating return results.
  - The **count** query parameter is supported for designating the maximum number of items in the return results.
  - The **meta** query parameter is supported with the **like** and the **favorite** values for requesting relationship metadata.

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one or more Post resources (the [“Post Resource” section on page 3-33](#) describes this resource).

**Example****Request**

```
GET http://webexsocialserver/api/quad/rest/communities/4710003/posts
Accept: application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
```

The response can also include the Post resource. For an example, see the [“Post Resource Response: JSON Format, Multiple Resources Returned”](#) section on page 18-64.

# Get Information about Questions and Answers for a Community

**Purpose** Retrieves information about questions and answers that were submitted when the designated community was created.

**Availability** Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
GET	{ <a href="#">http https</a> }:// <i>server</i> [: <i>port</i> ]/api/quad/rest/communities/ <i>community_id</i> /questionanswers[? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve questions and answers. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the QuestionAnswer resource to include in a response ([Table 3-40 on page 3-40](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter is not supported for sorting resources in a response
  - The **startIndex** query parameter is not supported for paginating return results
  - The **count** query parameter is not supported for designating the maximum number of items in the return results
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one or more QuestionAnswer resources (the [“QuestionAnswer Resource” section on page 3-40](#) describes this resource).

**Example****Request**

```
GET http://webexsocialserver/api/quad/rest/communities/4710003/questionanswers
Accept:application/json
Content-Type:application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "entry": [
    {
      "data": {
        "answer": "free form answer",
        "question": "This is the second question",
        "questionId": null
      }
    },
    {
      "data": {
        "answer": "answer1",
        "question": "This is the first question",
        "questionId": null
      }
    }
  ],
  "filtered": false,
  "itemsPerPage": 2,
  "serverMessages": null,
  "sorted": false,
  "startIndex": 0,
  "statusCode": 200,
  "totalResults": 0
}
```

# Get Information about the Document Library of a Community

**Purpose** Retrieves information about the document library of a designated community.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
GET	{ <a href="#">http</a> { <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/documentlibrary</a> [? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve document library information. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the DocumentLibrary resource to include in a response ([Table 3-21 on page 3-24](#) describes these fields).
  - The **filterBy** query parameter is not supported for filtering resources in a response.
  - The **sortOrder** query parameter does not apply.
  - The **startIndex** query parameter is supported for paginating return results by designating the first DocumentFolder or DocumentFile child resource that is returned. The DocumentFolder child resource takes precedence.
  - The **count** query parameter is supported for designating the maximum number of DocumentFolder and DocumentFile child resources that are included in the return results. DocumentFolder child resources take precedence.
  - The **meta** query parameter does not apply.

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one DocumentLibrary resource that contains representations of top-level document folders (the [“DocumentLibrary Resource” section on page 3-23](#) describes this resource).

**Example****Request**

```
GET http://webexsocialserver/api/quad/rest/1234/documentlibrary
Accept:application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
```

The response can also include the DocumentLibrary resource. For an example, see the [“DocumentLibrary Resource Response”](#) section on page 18-74.

# Get Information about Document Files in the Document Library of a Community

**Purpose** Retrieves information about all document files that are in the document library of the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
GET	{ <a href="#">http https</a> }:// <i>server</i> [: <i>port</i> ]/api/quad/rest/communities/ <i>community_id</i> /documentlibrary/@recent[? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community with the document library contains the document files about which information is to be retrieved. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the DocumentLibrary resource to include in a response ([Table 3-21 on page 3-24](#) describes these fields).
  - The **filterBy** query parameter is not supported for filtering resources in a response.
  - The **sortOrder** query parameter does not apply.
  - The **startIndex** query parameter is supported for paginating return results by designating the first DocumentFolder or DocumentFile child resource that is returned. The DocumentFolder child resource takes precedence.
  - The **count** query parameter is supported for designating the maximum number of DocumentFolder and DocumentFile child resources that are included in the return results. DocumentFolder child resources take precedence.
  - The **meta** query parameter does not apply.

**Request Payload** A request payload is not required.



---

**Response**

Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one DocumentLibrary resource that contains representations of all document files in the document library of the designated community (the [“DocumentLibrary Resource” section on page 3-23](#) describes this resource).

---

**Example****Request**

```
GET http://webexsocialserver/api/quad/rest/communities/1234/documentlibrary/@recent
Accept:application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
```

The response can also include the DocumentLibrary resource. For an example, see the [“DocumentLibrary Resource Response” section on page 18-74](#).

# Get Information about Document Files in the Document Library of a Community that are Authored by a Designated User

**Purpose** Retrieves information about document files that are in the document library of the designated community and that are authored by the designated user.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
GET	{ <a href="http://server[:port]/api/quad/rest/communities/community_id/documentlibrary?filterBy=author&amp;filterValue=user_id[?query_parameters]">http https</a> ://server[:port]/api/quad/rest/communities/community_id/documentlibrary?filterBy=author&filterValue=user_id[?query_parameters]}

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community library with the document library that contains the document files about which information is to be retrieved. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *user\_id*—Identifier of the user who authored the document files about which information is to be retrieved. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the DocumentLibrary resource to include in a response ([Table 3-21 on page 3-24](#) describes these fields).
  - The **filterBy** query parameter is required as shown in the URI.
  - The **sortOrder** query parameter does not apply.
  - The **startIndex** query parameter is supported for paginating return results by designating the first DocumentFolder or DocumentFile child resource that is returned. The DocumentFolder child resource takes precedence.
  - The **count** query parameter is supported for designating the maximum number of DocumentFolder and DocumentFile child resources that are included in the return results. DocumentFolder child resources take precedence.
  - The **meta** query parameter does not apply.

**Request Payload** A request payload is not required.

---

**Response**

Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one DocumentLibrary resource that contains representations of document files that are authored by the designated user (the [“DocumentLibrary Resource” section on page 3-23](#) describes this resource).

---

**Example****Request**

```
GET http://webexsocialserver/api/quad/rest/communities/1234/documentlibrary?
filterBy=author&filterValue=4321
Accept:application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
```

The response can also include the DocumentLibrary resource. For an example, see the [“DocumentLibrary Resource Response” section on page 18-74](#).

# Get Information about All Custom Attributes of a Community

**Purpose** Retrieves information about all custom attributes of the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
GET	{http https}://server[:port]/api/quad/rest/communities/community_id/customattributes[attribute[,...[attribute]]]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve information about custom attributes. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *attribute*—Name a custom attribute for which to retrieve information. When specifying multiple custom attribute names, separate each one with a comma.

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one or more Custom Attribute resources (the [“CustomAttribute Resource” section on page 3-19](#) describes this resource).

## Example

### Request

```
GET http://webexsocialserver/api/quad/rest/communities/1234/customattributes
Accept:application/json
```

### Response

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "statusCode": 200,
  "startIndex": 0,
  "itemsPerPage": 2,
  "totalResults": 2,
  "filtered": false,
  "sorted": false,
  "entry": [
    {
```

```
    "data": {
      "dataSelectionValues": [],
      "dataType": "String",
      "id": 0,
      "key": "Business",
      "resource": "http://quad.cisco.com/schema/1.0/customattribute",
      "value": ""
    }
  },
  {
    "data": {
      "dataSelectionValues": [],
      "dataType": "String",
      "id": 0,
      "key": "ExtendedDescription",
      "resource": "http://quad.cisco.com/schema/1.0/customattribute",
      "value": ""
    }
  }
],
"serverMessages": null
}
```

# Update Custom Attributes of a Community

**Purpose** Updates designated custom attributes for the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
PUT	{http https}://server[:port]/api/quad/rest/communities/community_id/customattributes/attributes

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to update custom attributes. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *attributes*—Comma-separated string of custom attributes to be updated. To retrieve the custom attributes that you can designate here, use the operation that the [“Get Information about All Custom Attributes of a Community” section on page 5-52](#) describes.

## Request Payload

The request payload includes the following fields from the CustomAttribute resource ([Table 3-16 on page 3-19](#) describes these fields). You must include one pair of key and value fields for each custom attribute that you specify in the request. These key/value pairs must be included in the same order as their corresponding custom attributes are specified in the request.

Required Fields	Optional Fields
key value	—

## Response

When updating a single custom attribute, standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only. When updating multiple custom attributes, standard response with a payload that includes the status of the updates.

## Example

### Request

```
PUT http://webexsocialserver/api/quad/rest/communities/310450/customattributes/Businesses
Accept: application/json
{
  "customAttribute": {
    "key": "Businesses",
```

```
        "value": "New value"  
    }  
}
```

**Response**

HTTP/1.1 204 No Content

# Get Information about Community Profile Images

**Purpose** Retrieves information about profile images for the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
GET	{ <a href="#">http</a> { <a href="#">https</a> }://server[:port]/api/quad/rest/communities/community_id/images/@profile[?query_parameters]}

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve the profile image. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the ProfileImage resource to include in a response ([Table 3-38 on page 3-39](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter is not supported for sorting resources in a response
  - The **startIndex** query parameter is not supported for paginating return results
  - The **count** query parameter is not supported for designating the maximum number of items in the return results
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include one or more ProfileImage resources (the [“ProfileImage Resource” section on page 3-39](#) describes this resource).

## Example

### Request

```
GET http://webexsocialserver/api/quad/rest/communities/4710003/images/@profile
Accept:application/json
Content-Type:application/json
```



**Response**

```
HTTP/1.1 200 OK  
Content-Type: application/json
```

The response can also include the ProfileImage resource. For an example, see the [“ProfileImage Resource Response”](#) section on page 18-75.

# Update a Community Profile Image

**Purpose** Updates the community profile image for the designated community.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
POST	{ <a href="#">http</a> / <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/images/@profile</a>

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to update the profile image. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities” section on page 1-12](#).

**Request Payload**

Image to be uploaded.

The HTTP Content-Type header must be multipart/form-data for this request. The form-data key must be specified as `image=image_to_upload`.

**Response**

Standard response (see the [“API Responses” section on page 1-17](#)) with an HTTP header only.

**Example**

**Request**

```
POST http://quadserver//api/quad/rest/communities/4710003/images/@profile
Accept:application/json
content-length: 87899

expect: 100-continue
content-type: multipart/form-data; boundary=-----02a4bb521c19
-----02a4bb521c19
Content-Disposition: form-data; name="image"; filename="ProfilePhoto.jpg"
<binary_file>
-----02a4bb521c19--
```

**Response**

```
HTTP/1.1 204 No Content
```

# Get All Activities that are Visible to You in a Community

## Purpose

Retrieves information about all activities that are visible to you in the designated community.

An activity consists of these components: *actor*, *action*, *object*, and *target*. An activity describes an action that is performed by an actor with or on an object that relates to a target. For example, an actor could be a Cisco WebEx Social user, an action could be “created,” an object could be a document, and a target could be a library.

A component may be explicit or implied. For example, an activity could be described as “Nicole updated a document in her library” or “Mike shared a post.”

## Availability

Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
GET	{ <a href="#">http</a> / <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/community_id/activities</a> [? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve information about activities that are visible to you. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities”](#) section on page 1-12.
- *query\_parameters*—Query parameters. See the [“Query Parameters”](#) section on page 1-20 for information about using query parameters. For this operation:
  - The **fields** query parameter does not apply.
  - The **filterBy** query parameter is supported in this format:  
**filterBy**={@**user**|@**community**|@**tag**}&**filterValue**=*value*.

For **filterBy**=@**user**, *value* is the identifier of a Cisco WebEx Social user. In this case, the operation retrieves only activities that include this user. You can use a *value* of **@me** to designate yourself. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities”](#) section on page 1-12.

For **filterBy**=@**community**, *value* is the identifier of a Cisco WebEx Social community. In this case, the operation retrieves only activities that include this community. For information about obtaining the identifier that you need, see the [“Identifiers of Cisco WebEx Social Entities”](#) section on page 1-12.

For **filterBy**=@**tag**, *value* is the name of a tag to an entity. In this case, the operation retrieves only activities that include this tag.

- The **sortOrder** query parameter is not supported for sorting resources in a response.
- The **startIndex** query parameter is not supported for paginating return results.

- The **count** query parameter is supported for designating the maximum number of items in the return results. The system retrieves activities in reverse chronological order, stopping when it retrieves the designated maximum number of activities or all activities, whichever number is smaller.
- The **meta** query parameter does not apply.
- The special query parameter **timestamp=***time\_to* is supported for retrieving activities that occurred before a designated date and time. Replace *time\_to* with the date and time, in UTC format, before which activities are to be retrieved.

**Request Payload**

A request payload is not required.

**Response**

Standard response (see the “[API Responses](#)” section on page 1-17) with a response payload that includes the following:

Item	Explanation
objectType	Appears as .
timestamp	Date and time that the earliest activity that is returned occurred, in UTC format.
totalItems	Total number of items that are returned.
items	Array of Activity activities resources. The “ <a href="#">Activity Activities Resource</a> ” section on page 3-48 describes this resource.

**Example****Request**

```
GET http://webexsocialserver/api/quad/rest/communities/1234/activities
Accept:application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
```

The response can also include the Collection activities resource. For an example, see the “[Collection Activities Resource Response](#)” section on page 18-121.

# Get Community Metrics

**Purpose** Retrieves metrics about various activities that occurred in a community.

**Availability** Cisco WebEx Social 3.3 and later.

**Request**

HTTP Method	URI
GET	{ <a href="#">http</a> { <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <b>api/quad/rest/community/community_id/communitymetrics</b> [? <b>startDate</b> = <i>start_date</i> & <b>endDate</b> = <i>end_date</i> ][? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *community\_id*—Identifier of the community for which to retrieve metrics. For information about obtaining the identifier that you need, see the “[Identifiers of Cisco WebEx Social Entities](#)” section on page 1-12.
- *start\_date*—Start of a date range for the metrics, in UTC format. For example 1332720000000 indicates March 26, 2012, 00:00:00 GMT. The earliest valid start date is 365 days before the current date.
- *end\_date*—End of a date range for the metrics, in UTC format. For example 1332806400000 indicates March 27, 2012, 00:00:00 GMT.
- *query\_parameters*—Query parameters. See the “[Query Parameters](#)” section on page 1-20 for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the communityMetrics resource to include in a response ([Table 3-11 on page 3-14](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter does not apply
  - The **startIndex** query parameter does not apply
  - The **count** query parameter does not apply
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response with a response payload that can include one CommunityMetrics resource (the “[CommunityMetrics Resource](#)” section on page 3-14 describes this resource).

---

**Examples****Request**

```
GET http://webexsocialserver/api/quad/rest/communities/1234/communitymetrics
?startDate=1359849600000&endDate=1359936000000
Accept:application/json
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "statusCode": 200,
  "startIndex": 0,
  "itemsPerPage": 1,
  "totalResults": 1,
  "filtered": false,
  "sorted": false,
  "entry": [
    {
      "data": {
        "id":100001,
        "startDate": 1359849600000,
        "endDate": 1359936000000,
        "membersJoined": 6,
        "membersLeft": 1,
        "postsShared": 43,
        "postsUnshared": 5,
        "commentsAdded": 12,
        "commentsRemoved": 1,
        "postLikesAdded": 4,
        "postLikesRemoved": 4,
        "commentLikesAdded": 4,
        "commentLikesRemoved": 4,
        "resource": "http://quad.cisco.com/schema/1.0/communitymetric"
      }
    }
  ],
  "serverMessages": null
}
```

# Get Code of Conduct

**Purpose** Retrieves the Cisco WebEx Social community code of conduct.

**Availability** Cisco WebEx Social 3.3 and later.

## Request

HTTP Method	URI
GET	{ <a href="#">http</a> { <a href="#">https</a> }:// <i>server</i> [: <i>port</i> ]/ <a href="#">api/quad/rest/communities/codeofconduct</a> [? <i>query_parameters</i> ]

where:

- *server*—Host name or IP address of the Cisco WebEx Social server.
- *port*—Port number on which the Cisco WebEx Social server listens for API requests. The default value is **80**.
- *query\_parameters*—Query parameters. See the [“Query Parameters” section on page 1-20](#) for information about using query parameters. For this operation:
  - The **fields** query parameter is supported for selecting fields from the CodeOfConduct resource to include in a response ([Table 3-8 on page 3-10](#) describes these fields)
  - The **filterBy** query parameter is not supported for filtering resources in a response
  - The **sortOrder** query parameter does not apply
  - The **startIndex** query parameter does not apply
  - The **count** query parameter does not apply
  - The **meta** query parameter does not apply

**Request Payload** A request payload is not required.

**Response** Standard response (see the [“API Responses” section on page 1-17](#)) with a response payload that can include the CodeOfConduct resource (the [“CodeOfConduct Resource” section on page 3-10](#) describes this resource).

## Example

### Request

```
GET http://webexsocialserver/api/quad/rest/communities/codeofconduct
Accept:application/json
Content-Type:application/json
```

### Response

```
HTTP/1.1 200 OK
Content-Type: application/json
{
```

```
"statusCode":200,
"startIndex":0,
"itemsPerPage":1,
"totalResults":0,
"filtered":false,
"sorted":false,
"entry":[
  {
    "data":{
      "enabled":false,
      "link":""
    }
  }
],
"serverMessages":null
}
```