

API

A guide for
managers &
developers

PART 6: Collections

Prepared by OfCourseMe

5. Collections

“Collections” service is the one that represents OfCourseMe the most and delivers the highest value to the final users.

The business need/use case goes like follows:

- When the user logs into the LMS / learning environment, I want to show her some custom playlists with free contents from the web, centered around our companies values / strategic plan / key challenges in the industry



This is the core of our “dynamic curation” service, which we explained in detail [here](#). [Here you can find a summary of the main steps:](#)

- 1.** The customer provides OfCourseMe with a detailed brief, including names of the collections, topics they want to cover, preferred format, providers, duration, languages
- 2.** OfCourseMe prepares the playlists and iterates together with the customer until they are satisfied with the outcome
OfCourseMe expose the playlists via APIs at the endpoint “Collections”
- 3.** The customer’s IT / system integrator uploads the collections onto the LMS / learning environment by calling the endpoint “Collections”
- 4.** After one quarter we look at the data – what users liked most, what else they looked for – and modify the playlists accordingly

When the user logs into the LMS / learning environment, she now sees the curated playlists immediately, wherever you decided to display them.

OfCourseMe APIs

5. Collections

Please find below the key basic information in order to get started, for developers

Description

Call endpoint <https://api.ofcourse.me/v2.0/collections>

This endpoint is useful to:

- get an array of pre-configured collections of courses (list of course_id)

To use this endpoint you don't need to pass any parameter.

You can pass a parameter "language" (two digit language code) to get only collections in the language needed.

You can pass a parameter "collection" (collection_id separated by comma) to get only collections needed.

Response Fields

Here are listed the response fields for a single collections:

- **collection_id**: OfCourseMe unique ID of the collection
- **collection_key**: OfCourseMe unique key of the collection (string)
- **collection_title**: title of the collection (string)
- **collection_audience_language**: audience language of the collection (string)
- **courses_ids**: a list of course_id (ordered by relevance) contained by collection (string)

Examples

<https://api.ofcourse.me/v2.0/collections>

Returns pre-configured collections of courses

<https://api.ofcourse.me/v2.0/collections?collection=1,2>

Returns pre-configured collections with collection_id 1 and 2

<https://api.ofcourse.me/v2.0/collections?language=en>

Returns pre-configured english collections of courses

<https://api.ofcourse.me/v2.0/collections?language=it>

Returns pre-configured italian collections of courses

Subset

[Learn more about subset and exclusive content](#)

Your calls could be limited by a defined subset on options basis.

This behavior affects also this endpoint: you will get only courses in your subset in returned collections.

User tracking parameters

It is possible, if desired, to identify a single user by appending a "user_id" parameter to your API-calls.

You can use a session id or an internal id (integer/string) to set the "user_id" parameter.

Also, if desired, you could identify a user group, family or entity by appending a "entity_id" parameter to your API-calls.

You can use an internal group, family or entity id (integer/string) to set the "entity_id" parameter.

This will help us computing API-usage statistics and metrics by single user and / or by user group / or by user family / or by user entity.

User tracking parameters are optionals.

Here is a basic example about the "/courses-search" endpoint.

https://api.ofcourse.me/v2.0/collections?&user_id=USER-ID

The "user_id" parameter is properly set to identify a single user

https://api.ofcourse.me/v2.0/collections?entity_id=GROUP-ID

The "entity_id" is properly set to identify user group

https://api.ofcourse.me/v2.0/collections?user_id=USER-ID&entity_id=GROUP-ID

The "user_id" parameter is properly set to identify a single user and "entity_id" is properly set to identify user group

https://api.ofcourse.me/v2.0/collections?&user_id=USER-ID&entity_id=FAMILY-ID

The "user_id" parameter is properly set to identify a single user and "entity_id" is properly set to identify user family

https://api.ofcourse.me/v2.0/collections?user_id=USER-ID&entity_id=ENTITY-ID

The "user_id" parameter is properly set to identify a single user and "entity_id" is properly set to identify user entity

Code Examples

[PHP \(CURL\)](#)

[C# \(WebRequest\)](#)

[Java \(URLConnection\)](#)

For testing purpose only

```
$.ajax({
  url: "https://api.ofcourse.me/v2.0/collections",
  type: GET,
  headers: {"x-api-key": "*****lcAuSy"} // insert your API Key
})
.done(function(response) {
  console.log(response);
})
.fail(function(err) {
  console.log(err);
});
```