



docebo® API MANUAL

www.docebo.com

Index

API Methods Description3

User Related API.....3

Course Related API11

Enrollment Related API22

Report Related API.....25

Course Document Related API26

API Manual

API Methods Description

The following methods are available through the instructions declared in the Technical Wiki manual.

User Related API

/api/user/checkUsername

Check if a user exists by its username (userid) or its email if also_check_as_email is true

Parameters:

Param name	Param type	Description
Userid	string	the userid to check
also_check_as_email	boolean	if true the userid will be checked also as an email address

Output example:

```
{
  "success": true,
  "idst": 123
}
```

/api/user/listUsers

Return a list of users; if "count" parameters is not specified, all records will be returned

Parameters:

Param name	Param type	Description
From	integer	
Count	integer	

Output example:

```
{
  "success": true,
  "users": [
    {
      "id_user": "12255",
      "userid": "user01",
      "firstname": "user",
      "lastname": "one",
      "email": "one@domain.com",
      "valid": "1"
    },
    {
      "id_user": "12250",
      "userid": "user02",
      ...
    }
  ]
}
```

/api/user/create

Create a new user through the API system; returns the idst of the new user.

Parameters:

Param name	Param type	Description
Userid	string	user userid (username)
Firstname	string	
Lastname	string	
Password	string	
Email	string	
reg_code	string	
reg_code_type	string	
ext_user_type	string	
ext_user	integer	
Role	string	
Language	string	
Orgchart	string	

Output example:

```
{
  "success": true,
  "idst": 12666
}
```

This function update a user trough the api system

Parameters:

Param name	Param type	Description
Idst	integer	numeric user id; not needed if using ext_user and ext_user_type
ext_user_type	string	(for example: drupal, joomla, ...)
ext_user	integer	numeric user id on the external platform
Firstname	string	
Lastname	string	
Password	string	
Email	string	
Valid	integer	0 = suspended / 1 = active

Output example:

```
{
  "success": true,
  "idst": "12666"
}
```

/api/user/delete

Delete a user trough the api system

Parameters:

Param name	Param type	Description
Idst	integer	numeric user id; not needed if using ext_user and ext_user_type
ext_user_type	string	(for example: drupal, joomla, ...)
ext_user	integer	numeric user id on the external platform

Output example:

```
{
  "success": true,
  "idst": "12666"
}
```

/api/user/suspend

Suspend a user trough the api system

Parameters:

Param name	Param type	Description
Idst	integer	numeric user id; not needed if using ext_user and ext_user_type
ext_user_type	string	(for example: drupal, joomla, ...)
ext_user	integer	numeric user id on the external platform

Output example:

```
{
  "success": true,
  "idst": "12666"
}
```

/api/user/unsuspend

Unsuspend / reactivate a user trough the api system

Parameters:

Param name	Param type	Description
Idst	integer	numeric user id; not needed if using ext_user and ext_user_type
ext_user_type	string	(for example: drupal, joomla, ...)
ext_user	integer	numeric user id on the external platform

Output example:

```
{
  "success": true,
  "idst": "12666"
}
```

/api/user/userCourses

Return the list of courses user is subscribed to

Parameters:

Param name	Param type	Description
Idst	integer	numeric user id; not needed if using ext_user and ext_user_type
ext_user_type	string	(for example: drupal, joomla, ...)
ext_user	integer	numeric user id on the external platform

Output example:

```
{
  "0": {
    "course_info": {
      "course_id": "9",
      "course_name": "Business english Telephone skills",
      "course_description": "",
      "can_enter": {
        "can": true,
        "reason": "course_status",
        "expiring_in": false
      },
      "course_link":
"http://mylms.xyz/doceboLms/index.php?modname=course&op=aula&idCourse=9"
    }
  },
  "success": true
}
```

/api/user/associateExternalUsers

Associate a list of external user id and type with the corresponding Docebo users

Parameters:

Param name	Param type	Description
from_email	boolean	compare with the email instead of userid
Userdata	array	

Parameters example (PHP):

```
array(  
    'from_email'=>false,  
    'userdata[user_1][idst]'=>12667,  
    'userdata[user_1][ext_user_type]'=>'test',  
    'userdata[user_1][ext_user]'=>123,  
    'userdata[user_2][userid]'=>'username 02',  
    'userdata[user_2][ext_user_type]'=>'test',  
    'userdata[user_2][ext_user]'=>124,  
);
```

Output example:

```
{  
    "success": true,  
    "sync_12667": "username",  
    "sync_271": "username 02"  
}
```

/api/auth/authenticate

Check if valid username and password are provided and returns the authentication token required for single sign on

Parameters:

Param name	Param type	Description
Username	string	the user userid
Password	string	

Output example:

```
{
  "success": true,
  "message": "You are authenticated.",
  "idst": "123",
  "token": "6b8039457e64dca94cac0e73d3708034",
  "expire_at": "2011-11-10 15:12:35"
}
```

Course Related API

/api/course/listCourses

Get the list of all available courses, eventually filtering results according to a given category

Parameters:

Param name	Param type	Description
Category	integer	

Output example:

```
{
  "0": {
    "course_info": {
      "course_id": "34",
      "code": "123456789",
      "course_name": "Course name",
      "course_description": "Description",
      "status": "2",
      "selling": "0",
      "price": "",
      "subscribe_method": "2",
      "course_edition": "0",
      "course_type": null,
      "sub_start_date": null,
      "sub_end_date": null,
      "date_begin": "2010-10-12",
      "date_end": "2010-10-13",
      "course_link":
"http://my.lms/doceboLms/index.php?modname=course&op=aula&idCourse=34"
    }
  },
  "1": {
    "course_info": {
      "course_id": "23",
      "code": "00001",
      ...
    }
  }
}
```

```
}  
  
},  
  
"success": true  
  
}
```

/api/course/listEditions

Get the list of all the editions of a given course

Parameters:

Param name	Param type	Description
course_id	integer	(not needed if using course_code)
course_code	string	

Output example:

```
{  
  "0": {  
    "course_info": {  
      "course_id": "40",  
      "edition_id": "7",  
      "code": "edition01_01",  
      "course_name": "Course Edition 01",  
      "course_description": "",  
      "status": "2",  
      "selling": "0",  
      "price": "",  
      "subscribe_method": "2",
```

```
    "sub_start_date": "0000-00-00",
    "sub_end_date": "0000-00-00",
    "date_begin": "0000-00-00",
    "date_end": "0000-00-00",
    "course_link":
"http://my.lms/doceboLms/index.php?modname=course&op=aula&idCourse=40"
  }
},
"1": {
  "course_info": {
    "course_id": "40",
    "edition_id": "8",
    "code": "edition01_02",
    ...
  }
},
"success": true
}
```

/api/course/listClassrooms

Get the list of all the classrooms of a given course

Parameters:

course_id (not needed if using course_code)

course_code

Param name	Param type	Description
------------	------------	-------------

course_id	integer	not needed if using course_code
course_code	string	

Output example:

```
{
  "0": {
    "course_info": {
      "course_id": "41",
      "date_id": "23",
      "code": "classroom01_01",
      "course_name": "Classroom course - class 01",
      "course_description": "",
      "status": "0",
      "selling": "0",
      "price": "15",
      "subscribe_method": "2",
      "sub_start_date": "0000-00-00 00:00:00",
      "sub_end_date": "0000-00-00 00:00:00",
      "date_begin": "2011-01-14 01:05:00",
      "date_end": "2011-02-26 00:00:00",
      "num_day": "4",
      "classroom": "Class 01",
      "course_link":
"http://my.lms/doceboLms/index.php?modname=course&op=aula&idCourse=41"
    }
  },
  "1": {
    "course_info": {
      "course_id": "41",
      "date_id": "24",
```

```
    "code": "classroom01_02",  
    ...  
  }  
},  
"success": true  
}
```

/api/course/addCourse

<description>

Parameters:

Param name	Param type	Description
Advance	string	
allow_overbooking	boolean	
auto_unsubscribe	boolean	
can_subscribe	integer	
course_advanced	boolean	
course_autoregistration_code	string	
course_code	string	
course_date_begin	string	
course_date_end	string	

course_day_of	integer	
course_descr	string	
course_difficult	string	veryeasy,easy,medium,difficult,verydifficult
course_edition	boolean	if course will use editions
close_lo_perm	boolean	
course_lang	string	language code (english,italian,...)
course_medium_time	string	
course_name	string	
course_price	string	
course_progress	boolean	show course progress?
course_quota	string	
course_sell	boolean	is the course for sale?
course_show_rules	integer	
course_sponsor_link	string	
course_status	integer	
subscribe_method	integer	
show_course_time	boolean	
course_type	string	
Credits	integer	
direct_play	boolean	enable course auto play?
category_id	integer	

inherit_quota	boolean	
max_num_subscribe	integer	
min_num_subscribe	integer	
random_course_autoregistration_code	string	
show_result	boolean	
show_who_online	boolean	
sub_end_date	string	subscriptions end date
sub_start_date	string	subscriptions start date
unsubscribe_date_limit	string	
use_logo_in_courselist	boolean	
use_unsubscribe_date_limit	boolean	

Output example:

```
{
  "success": true,
  "course_id": "45"
}
```

/api/course/updateCourse

Update course details

Parameters:

Param name	Param type	Description
course_id	integer	the numeric id of the course

(for other parameters, see the addCourse parameters table)

Output example:

```
{
  "success": true,
  "idst": "12666"
}
```

/api/course/deleteCourse

Delete a course

Parameters:

Param name	Param type	Description
course_id	integer	

Output example:

```
{
  "success": true
}
```

/api/user/addClassroom

Add a classroom edition to a course

Parameters:

Param name	Param type	Description
course_id	integer	id of the parent course
Code	string	
Name	string	
Description	string	
medium_time	integer	
max_users	integer	
Price	string	
allow_overbooking	boolean	
Status	integer	
test_type	integer	
sub_start_date	string	
sub_end_date	string	
unsubscribe_date_limit	string	

Output example:

```
{
  "success": true,
  "classroom_id": "16"
}
```

/api/user/updateClassroom

Update a course classroom edition

Parameters:

Param name	Param type	Description
classroom_id	integer	id of the classroom
Code	string	
Name	string	
Description	string	
medium_time	integer	
max_users	integer	
Price	string	
allow_overbooking	boolean	
Status	integer	
test_type	integer	
sub_start_date	string	
sub_end_date	string	
unsubscribe_date_limit	string	

Output example:

```
{
  "success": true
}
```

/api/user/delClassroom

Delete a course classroom edition

Parameters:

Param name	Param type	Description
classroom_id	integer	id of the classroom

Output example:

```
{  
  "success": true  
}
```

Enrollment Related API

/api/course/addUserSubscription

Subscribe an user to a course, edition or classroom

Parameters:

Param name	Param type	Description
Idst	integer	
course_id	integer	
course_code	string	
edition_id	integer	

edition_code	string	
classroom_id	integer	
classroom_code	string	
user_level	string	

Output example:

```
{
  "success": true,
  "message": "User has been subscribed to the course"
}
```

/api/course/updateUserSubscription

Update the subscription of a given user related to a course, edition or classroom

Parameters:

Param name	Param type	Description
Idst	integer	
course_id	integer	
course_code	string	
edition_id	integer	
edition_code	string	
classroom_id	integer	

classroom_code	string	
user_level	string	
user_status	string	

Output example:

```
{
  "success": true,
  "message": "User subscription has been updated"
}
```

/api/course/deleteUserSubscription

Delete user subscription form a course, edition or classroom

Parameters:

Param name	Param type	Description
Idst	integer	
course_id	integer	
course_code	string	
edition_id	integer	
edition_code	string	
classroom_id	integer	
classroom_code	string	

Output example:


```
{
  "success": true,
  "message": "User has been removed from the course"
}
```

/api/course/subscribeUserWithCode

Subscribe a user to a course using a registration code

Parameters:

Param name	Param type	Description
Idst	integer	
reg_code_type	string	
reg_code	string	

Output example:

```
{
  "success": true,
  "message": "User has been subscribed to the course"
}
```

Report Related API

/api/report/listReports

Return a list of all available reports

Parameters:

Param name	Param type	Description
------------	------------	-------------

only_public	Boolean	show only public reports
-------------	---------	--------------------------

Output example:

```
{
  "success": true,
  "report_list": {
    "8": {
      "id_filter": "8",
      "id_report": "2",
      "author": "1424",
      "creation_date": "2010-08-21 11:13:14",
      "filter_name": "Users - Competences",
      "is_public": "1",
      "views": "15",
      "idst": "1424",
      "userid": "user.name"
    },
    "9": {
      "id_filter": "9",
      "id_report": "2",
      ...
    }
  }
}
```

Course Document Related API

/api/learningObj/loadCloudScorm

Load a new cloud front stored scorm object

Parameters:

Param name	Param type	Description
course_id	Integer	not needed if using course_code
course_code	String	
obj_name	String	
obj_address	String	address on cloudfront

Output example:

```
{  
  "success": true  
}
```

/api/learningObj/loadScorm

Load a new scorm from a zip file located on a remote server

Parameters:

Param name	Param type	Description
course_id	Integer	not needed if using course_code
course_code	String	
obj_name	String	
zip_address	String	url of the zip archive

Output example:

```
{
  "success": true
}
```

/api/learningObj/listCourseObjects

List all learning objects of a course

Parameters:

Param name	Param type	Description
course_id	Integer	numeric id of the course

Output example:

```
{
  "success": true,
  "object_list": {
    "51": {
      "scorm_id": "51",
      "parent_id": "0",
      "title": "My Scorm",
      "object_type": "scormorg",
      "resource_id": "21",
      "category_id": "0",
      "description": "",
      "language": ""
    }
  }
}
```

```
    "course_id": "45",
    "prerequisites": "",
    "is_terminator": "1",
    "publish_from": null,
    "publish_to": null
  },
  "52": {
    "scorm_id": "52",
    ...
  },
}
```

/api/learningObj/setObjectProperties

Set learning object common properties

Parameters:

Param name	Param type	Description
scorm_id	Int	
is_terminator	Boolean	set if the object is the end object marker
bookmark	String	values: "-", "start", "end"

Output example:

```
{
  "success": true,
  "idst": "12666"
}
```

/api/learningObj/create Test

Create a new test learning object

Parameters:

Param name	Param type	Description
course_id	integer	not needed if using course_code
course_code	String	
obj_name	String	
description		
point_type		number or percentage
point_required		
display_type		
order_type		
shuffle_answer		
question_random_number		
allow_resume		
allow_edit_answer		
allow_page_navigation		
show_answers		show all the answers inserted by user
show_correct		show correct answers

max_attempt		
hide_info		hide info for test in use
use_suspension		use test suspension
suspension_num_attempts	integer	suspend the test after N number of failed attempts
suspension_num_hours	integer	suspend the test for the N number of hours
suspension_prerequisites	boolean	after the test suspension, require a new completion of all the test prerequisites
mandatory_answer	boolean	ask the user to answer all questions

Output example:

```
{
  "success": true,
  "scorm_id": "54"
}
```

/api/learningObj/importTestQuestions

Import questions and answers of a quest from a given GIFT file / string

Parameters:

Param name	Param type	Description
scorm_id	Integer	
gift_url	String	the url of the GIFT file to import
gift_text	String	the content of the GIFT file to import; not needed if using gift_url

Output example:

```
{
  "success": true,
  "imported": [
    {
      "quest_type": "Single Choice",
      "success": 7,
      "fail": ""
    }
  ]
}
```