

# API Reference

This document contains information about the application programming interfaces (APIs) offered by Educatia AS to developers. Note that you need a valid license to use the web services described here.

## General

The API uses a REST-style interface over HTTP(S). The API endpoint for the cloud service is `https://secure.e-learning.no/web/services` and all paths below are relative to that. Different endpoints will exist for on-premise installations and for test environments.

Developers may use command-line tools such as `curl` or browser plugins such as `RESTClient` for Firefox to test the API.

## API Keys and Authentication

The API is stateless and each request must contain an API key to authenticate the request. An API key is associated with a specific user account. The API key has the same permissions and privileges as the user when that user is logged into the system via the GUI. It is recommended that you create a separate user for API usage, and assign this user the minimum privileges it needs to do its work.

### How to get your API Key

To view the API key of a user, do the following:

1. Login as the System Administrator
2. Click "Administrators"
3. Click on an administrator (or create one if one does not already exist)
4. Click the "Change" button
5. Set the "Shared Account" setting to "Yes" and click "Save Changes"
6. On the Administrator Details page, click the "View API Key" link to view the API Key

**Note:** API keys should be treated like passwords and should never be exposed in client-side code (such as Javascript) and never transmitted in plain text. All web service calls should be over HTTPS to ensure that the API key is protected.

If an API key has been compromised, a new API key can be issued by the system administrator by changing the password for the user associated with the API key.

All requests must be made over HTTPS (HTTP requests will return an error). Note that test environments may be configured to accept insecure (HTTP) requests to facilitate testing and troubleshooting.

## Request Format

The API receives values via HTTP requests. Parameter values are passed using HTTP POST.

A sample request looks like this:

```
1. curl https://secure.e-learning.no/web/services/exam.get_results -d "api_key=YOUR_API_KEY&exam_session_id=1234"
```

In addition to parameters specific to each service operation, the following parameters are common to all calls:

Parameter	Description	Required	Default
api_key	The API key used to authenticate. See the section about API Keys above for more information.	Yes	
api_version	Specifies the API version used to make the call. At the time of writing, version 1.0 is the latest version.	No	<i>latest_version</i>

For brevity, these parameters are not included in the example calls below.

## Response Format

The API always returns a JSON object with the Content-Type set to `application/json`. If the request was successful, the requested object is returned. If the call fails, a JSON object is returned with details about the error.

Unless otherwise noted, date values are represented using the ISO 8601 ([http://en.wikipedia.org/wiki/ISO\\_8601](http://en.wikipedia.org/wiki/ISO_8601)) format, as local time (without a time zone designator), for example `2014-01-20T12:57:00`. Booleans are represented with `Y` for true and `N` for false.

Example Response:

```
1. [{
2.   "exam_session_status": "COMPLETED",
3.   "exam_session_result": "FAILED",
4.   "exam_session_score": "60",
5.   "api_status": "OK",
6.   "api_message": "Session found.",
7.   "api_datetime": "2014-01-20T12:57:00"
8. }]
```

In addition to return values specific to each service operation, the following return values are common to all calls:

Property	Description
api_status	A code indicating the status of the call. If the call was successful, the string <code>OK</code> is

	returned, otherwise the string <code>ERROR</code> is returned.
<code>api_message</code>	A text string containing more information about the result of the call.
<code>api_datetime</code>	The current server date and time.

For brevity, these properties are not included in the example return values below.

# Person Service

## List persons

This call lists persons (users).

`/person.list`

The call has the following parameters:

Parameter	Description	Required	Default
<code>search_name</code>	A string that filters persons by name.	No	
<code>search_email</code>	A string that filters persons by email address.	No	
<code>search_group_name</code>	A string that filters persons by group name.	No	

The call returns the following:

Property	Description
<code>list</code>	A list of persons matching the search criteria.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.list -d "search_name=john"
```

Example Response:

```
1. [{
2. "list":[
3. {
4. "group_name":"Sales",
5. "person_email":"kari@example",
6. "person_id":"1342",
7. "person_name":"Kari Johnsen"
8. },
9. {
10. "group_name":"Sales",
11. "person_email":"john@example",
12. "person_id":"259",
13. "person_name":"John Doe"
14. },
15. {
16. "group_name":"Sales",
17. "person_email":"",
18. "person_id":"1529",
19. "person_name":"Johnny Mueller"
20. }
21. ]
22. }]
```

## Get temporary login URL

This call returns a temporary login link for a given person (user). It can be used to enable Single Sign-On from your system, so that users do not have to login with a username and password.

For example, if you want users to access our service from your corporate intranet, you would make a call to this API and pass the ID of the person (if necessary, use the `person.list` API first to retrieve the person ID based on email or name). Alternatively, pass one or more of the alternative identifiers (employee number, mobile phone number, etc.) to identify a person.

Then you simply redirect to the URL returned from the API (or add the URL as a link that can be clicked by the user on a web page, or perhaps in an email).

Note that the URL changes each time this API is called, and expires when the URL is used to login, so do not store the URL permanently. You should request a new, temporary login URL each time the user wants to login.

**`/person.get_temporary_login_url`**

The call has the following parameters:

--	--	--	--

Parameter	Description	Required	Default
person_id	A number that identifies a person.	No	
person_name	The name of the person.	No	
person_email	The email address of the person.	No	
person_mob	The mobile phone number of the person.	No	
employee_number	The employee number of the person.	No	

The call returns the following:

Property	Description
person_login_url	A unique link that allows the user to login (one time) without specifying a username or password.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.get_temporary_login_url -d "person_id=1234"
```

Example Response:

```
1. [{
2. "person_login_url":"https://secure.e-learning.no/sso/example-company/1234/ILQ4AM1B2T9ONO
D3E8MRNSAZXKJWCI50LZH5"
3. }]
```

## Add person

This call adds a new person (user).

**/person.add**

The call has the following parameters:

Parameter	Description	Required	Default
person_name	The name of the person.	Yes	
person_email	The email address of the	No	

	person.		
group_id	A number that identifies a group.	No	<i>The group defined as default for new users.</i>

The call returns the following:

Property	Description
person_id	A number that identifies the person.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.add -d "person_name=john"
```

Example Response:

```
1. [{
2. "person_id": "1234"
3. }]
```

## Add or update person

This call adds a new person (user) or updates an existing person (user). This API call is typically used to keep user data synchronized with an external system. When trying to determine if the person already exists, the system first uses the person ID (if specified), then the employee number, and finally attempts to match against the person's name, email address and phone number.

**/person.add\_or\_update**

The call has the following parameters:

Parameter	Description	Required	Default
person_id	A number that identifies a person.	No	
person_name	The name of the person.	Yes	
person_email	The email address of the person.	No	
person_mob	The mobile phone number of the person.	No	
employee_number	The employee number of the person.	No	

company_name	The name of the company or organization that the person works for.	No	
department_name	The name of the department that the person belongs to.	No	
job_title	The job title (position) of the person.	No	
profile_list	A colon-separated list of competency profiles that should be assigned to the person.	No	
group_id	A number that identifies a group that the new person should be added to (new persons only). If specified, the group ID takes precedence over the group code and group name.	No	<i>The group defined as default for new users.</i>
group_code	A string that identifies a group that the new person should be added to (new persons only). If specified, the group code takes precedence over the group name.	No	
group_name	The name of the group that the new person should be added to (new persons only).	No	
lang_code	The language code of the person.	No	
manager_number	The employee number of the person's manager.	No	
is_active	A boolean that indicates if the person is active or not. Inactive users will not be able to login to the system.	No	Y
send_message	A boolean indicating whether the person should receive a welcome message (new persons only).	No	N
employed_from_date	The person's employment start date.	No	
employed_to_date	The person's employment to date.	No	
employment_category	The person's employment category (full time, part time, contractor, apprentice, retired, etc.).	No	
employment_pct	The person's employment as a percentage of full-time employment.	No	
salary_group	The person's salary group code.	No	

salary_level	The person's salary level.	No	
gender	The person's gender. Valid values are M for males or F for females.	No	
birth_date	The person's birth date.	No	
nationality	The person's nationality.	No	
next_of_kin_name	The person's next of kin.	No	
address	The person's postal address.	No	
post_code	The person's postal code.	No	
post_area	The person's postal area name.	No	
country_name	The person's country of residence.	No	

The call returns the following:

Property	Description
person_id	A number that identifies the person.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.add_or_update -d "person_name=john&employee_number=4567&department_name=Accounting"
```

Example Response:

```
1. [{
2. "person_id": "1234"
3. }]
```

## Move person to trashcan

This call moves a person to the trashcan (Recycle Bin). To permanently delete a person's data, an administrator must purge the trashcan.

**/person.trash**

The call has the following parameters:

--	--	--	--

Parameter	Description	Required	Default
person_id	A number that identifies a person.	No	
person_name	The name of the person.	No	
person_email	The email address of the person.	No	
person_mob	The mobile phone number of the person.	No	
employee_number	The employee number of the person.	No	

The call returns the following:

Property	Description
person_id	A number that identifies the person.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.trash -d "employee_number=4567"
```

Example Response:

```
1. [{
2. "person_id":"1234"
3. }]
```

## List person messages

This call lists all messages sent to a given person.

**/person.list\_messages**

The call has the following parameters:

Parameter	Description	Required	Default
person_id	A number that identifies a person.	Yes	

The call returns the following:

Property	Description

list

A list of messages.

**Example Request:**

```
1. curl https://secure.e-learning.no/web/services/person.list_messages -d "person_id=1234"
```

**Example Response:**

```
1. [{
2. "list":[
3. {
4. "body":"Click this link to login: \n\nNOTE: The link above contains your private code,
do not share the link with others.\n\nIf you need help, please contact no-reply@exampl
e",
5. "message_id":"3541",
6. "sent_date":"",
7. "title":"Welcome to e-learning at ExampleCompany",
8. "type":"SEC"
9. },
10. {
11. "body":"Click this link to start: \n\nNOTE: You must complete "Your first course" by 2
0.01.2014",
12. "message_id":"3542",
13. "sent_date":"",
14. "title":"Your first course",
15. "type":"NOT"
16. }
17. ]
18. }]
```

## List person courses

This call lists all courses a given person has been enrolled in.

**/person.list\_courses**

The call has the following parameters:

Parameter	Description	Required	Default
person_id	A number that identifies a person.	Yes	

The call returns the following:

--	--

Property	Description
list	A list of courses.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.list_courses -d "person_id=1234"
```

Example Response:

```
1. [{
2. "list":[
3. {
4. "completed_date": "",
5. "course_enrolment_id": "850",
6. "course_name": "Your first course",
7. "feedback": "",
8. "progress": "0",
9. "rating": "",
10. "started_date": ""
11. }
12. ]
13. }]
```

## Add a course to person's course history

This call adds a course to a person's history of courses.

**/person.add\_course\_history**

The call has the following parameters:

Parameter	Description	Required	Default
person_id	A number that identifies a person. If this is specified, it takes precedence over the employee number.	No	
employee_number	A string that identifies a person.	No	
course_id	A number that identifies a course. If this is specified, it takes precedence over the course code.	No	
course_code	A string that identifies a course.	No	
start_date	The start date of the course.	No	

end_date	The end date of the course.	No	
expiry_date	The expiry date of the course/certificate.	No	
diploma_url	A link (URL) to the person's diploma or other documentation of the completed course.	No	
location_name	The name of the location where the course was taken.	No	
teacher_name	The name of the course instructor.	No	
remarks	Comments related to the course.	No	
skip_duplicates	A boolean that indicates if existing person/course combinations should be skipped.	No	Y

The call returns the following:

Property	Description
course_session_id	A number that uniquely identifies the course session.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.add_course_history -d "course_id=100&person_id=991"
```

Example Response:

```
1. [{
2.   "course_session_id": "1234"
3. }]
```

## List person exams

This call lists all exams a given person has started or completed.

**/person.list\_exams**

The call has the following parameters:

Parameter	Description	Required	Default
person_id	A number that identifies a person.	Yes	

The call returns the following:

Property	Description
list	A list of exams.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.list_exams -d "person_id=1234"
```

Example Response:

```
1. [{
2. "list":[
3. {
4. "completed":"Y",
5. "completed_date":"2014-01-19T16:56:15",
6. "exam_name":"Computer Skills Assessment",
7. "exam_session_id":"830",
8. "passed":"N",
9. "score":"0",
10. "started_date":"2014-01-19T16:56:04"
11. },
12. {
13. "completed":"Y",
14. "completed_date":"2014-01-19T16:56:29",
15. "exam_name":"Office 2010 Pre-Test",
16. "exam_session_id":"831",
17. "passed":"Y",
18. "score":"100",
19. "started_date":"2014-01-19T16:56:21"
20. }
21. ]
22. }]
```

## List person CV items

This call lists all CV items for given person.

```
/person.list_cv
```

The call has the following parameters:

Parameter	Description	Required	Default
-----------	-------------	----------	---------

person_id	A number that identifies a person.	Yes	
-----------	------------------------------------	-----	--

The call returns the following:

Property	Description
list	A list of CV items.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.list_cv -d "person_id=1234"
```

Example Response:

```
1. [{
2.   "list":[
3.     {
4.       "description":"","
5.       "from_date":"","
6.       "item_id":"261",
7.       "location":"","
8.       "name":"English",
9.       "organization":"","
10.      "to_date":"","
11.      "type":"LAN"
12.    },
13.    {
14.      "description":"","
15.      "from_date":"","
16.      "item_id":"262",
17.      "location":"","
18.      "name":"German",
19.      "organization":"","
20.      "to_date":"","
21.      "type":"LAN"
22.    },
23.    {
24.      "description":"","
25.      "from_date":"","
26.      "item_id":"263",
27.      "location":"","
28.      "name":"French",
29.      "organization":"","
30.      "to_date":"","
31.      "type":"LAN"
32.    },
33.    {
34.      "description":"","
35.      "from_date":"2014-01-01T00:00:00",
36.      "item_id":"264",
37.      "location":"","
38.      "name":"Certified Technician",
39.      "organization":"","
40.      "to_date":"2014-12-31T00:00:00",
41.      "type":"CER"
42.    }
43.  ]
44. }]
```

## Search CVs

This call searches for persons (users) with CVs that contain matching keywords.

**/person.search\_cv**

The call has the following parameters:

Parameter	Description	Required	Default
search_string	A string that represents the keywords being searched for.	Yes	
group_id	A number that identifies a group. If specified it limits the search to persons in the group.	No	
since_days	A number that limits the search to recent CV entries.	No	

The call returns the following:

Property	Description
list	A list of persons matching the search criteria.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/person.search_cv -d "search_string=true&group_id=1&since_days=30"
```

Example Response:

```
1. [{
2. "list":[
3. {
4. "person_id":"1342",
5. "person_name":"Kari Johnsen",
6. "search_score":"12"
7. },
8. {
9. "person_id":"259",
10. "person_name":"John Doe",
11. "search_score":"7"
12. },
13. {
14. "person_id":"1529",
15. "person_name":"Johnny Mueller",
16. "search_score":"3"
17. }
18. ]
19. }]
```

# Course Service

## List courses

This call lists courses.

**/course.list**

The call has the following parameters:

Parameter	Description	Required	Default
search_name	A string that filters courses by name.	No	
search_subject	A string that filters courses by subject.	No	
course_type	A string that filters courses by type. Valid values are STD for standard e-learning courses, CLA for classroom courses, and EXT for external courses.	No	

The call returns the following:

Property	Description
list	A list of courses matching the search criteria.

Example Request:

```
1. curl https://secure.e-Learning.no/web/services/course.list -d "search_type=EXT"
```

Example Response:

```

1. [{
2.   "list":[
3.     {
4.       "course_id":"114",
5.       "course_name":"Safety Course 101",
6.       "course_subject":"HSEQ"
7.     },
8.     {
9.       "course_id":"200",
10.      "course_name":"Fire Prevention Course",
11.      "course_subject":"HSEQ"
12.     }
13.   ]
14. }]}

```

## Add course

This call adds a new course.

**/course.add**

The call has the following parameters:

Parameter	Description	Required	Default
course_name	The name of the course.	Yes	
course_code	A string that identifies the course.	No	
course_url	A link (URL) with more information about the course.	No	
course_description	A description of the course.	No	
folder_id	A number that identifies a folder location for the course.	No	

The call returns the following:

Property	Description
course_id	A number that identifies the course.

Example Request:

```
1. curl https://secure.e-Learning.no/web/services/course.add -d "course_name=HSEQ"
```

Example Response:

```
1. [{
2.   "course_id": "1234"
3. }]
```

## Enrol person in course

This call enrolls a person in a course and returns information about the enrolment.

**/course.enrol\_person**

The call has the following parameters:

Parameter	Description	Required	Default
person_id	A number that identifies a person (student).	No	
course_id	A number that identifies a course.	No	
send_message	A boolean indicating whether the person should receive a message notification about the enrolment.	No	Y

The call returns the following:

Property	Description
course_session_id	A number that uniquely identifies the course session (enrolment).
course_session_url	A unique link (URL) that can be used to view the course.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/course.enrol_person -d "course_id=100&person_id=991"
```

Example Response:

```
1. [{
2.   "course_session_id": "1234",
3.   "course_session_url": "https://secure.e-learning.no/course/example-company/1234/ILQ4AM1B2T90NOD3E8MRNSAZXKJWCII50LZH5"
4. }]
```

## Create standalone course session

This call creates a "standalone" session; that is, it registers a new student and grants access to a single course. It returns a unique URL used to start that course, and an course session ID that can be used to track course session status.

**/course.create\_standalone\_session**

The call has the following parameters:

Parameter	Description	Required	Default
course_id	A number that identifies a course.	Yes	
student_name	The name of the student.	No	
student_email	The email of the student. If this is specified, the student will receive an email which includes the link to start the course.	No	
exit_url	A URL that the student will be redirected to when the course is completed. If specified, this session-specific URL overrides any fixed exit URL configured for the course.	No	

The call returns the following:

Property	Description
person_id	A number that uniquely identifies the person (student).
course_session_id	A number that uniquely identifies the course session.
course_session_url	A unique link (URL) that can be used to start the course.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/course.create_standalone_session -d "course_id=1234&student_name=john%20doe"
```

Example Response:

```
1. [{
2.   "course_session_id": "1234",
3.   "course_session_url": "https://secure.e-learning.no/course/example-company/1234/ILQ4AM1B2T90NOD3E8MRNSAZXKJWCI50LZH5",
4.   "person_id": "5551"
5. }]
```

## Get course enrolment status

This call retrieves the course status results for a given course session (enrolment).

**/course.get\_enrolment\_status**

The call has the following parameters:

Parameter	Description	Required	Default
course_session_id	A number that identifies a course session (enrolment).	Yes	

The call returns the following:

Property	Description
course_session_id	A number that identifies a course session (enrolment).
course_session_progress	A number that represents the progress of the course session, as a percentage of the total number of pages in the course.
course_session_status	A string that indicates the status of the exam session. Possible values are APPROVAL_REQUIRED , ENROLLED , IN_PROGRESS , COMPLETED , OBSOLETE or CANCELLED .

Example Request:

```
1. curl https://secure.e-learning.no/web/services/course.get_enrolment_status -d "course_session_id=1234"
```

Example Response:

```
1. [{
2.   "course_session_id": "1234",
3.   "course_session_progress": "100",
4.   "course_session_status": "COMPLETED"
5. }]
```

## Exam Service

### List exams

This call lists exams.

**/exam.list**

The call has the following parameters:

Parameter	Description	Required	Default
search_name	A string that filters exams by name.	No	
search_subject	A string that filters exams by subject.	No	

The call returns the following:

Property	Description
list	A list of exams matching the search criteria.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/exam.list -d "search_subject=office"
```

Example Response:

```
1. [{
2. "list":[
3. {
4. "exam_id":"114",
5. "exam_name":"Advanced Macros with Excel 2010",
6. "exam_subject":"MS Office"
7. },
8. {
9. "exam_id":"200",
10. "exam_name":"Office 2010 Pre-Test",
11. "exam_subject":"MS Office"
12. }
13. ]
14. }]
```

## Grant exam access

This call grants access to an exam and returns a unique URL used to start that exam.

**/exam.grant\_access**

The call has the following parameters:

Parameter	Description	Required	Default
person_id	A number that identifies a person (student).	Yes	
exam_id	A number that identifies an exam.	Yes	
access_ip_address	If this value is specified, access to the exam is restricted to this IP address.	No	
access_duration	The duration of access to the exam. The value is specified in minutes.	No	60

The call returns the following:

Property	Description
exam_access_url	A unique link (URL) that can be used to start the exam. The link is valid for a specific duration, and for a specific IP address if that was specified.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/exam.grant_access -d "exam_id=432&person_id=991"
```

Example Response:

```
1. [{
2.   "exam_access_url": "https://secure.e-learning.no/exam/example-company/1234/ILQ4AM1B2T9ONOD3E8MRNSAZXKJWC150LZH5"
3. }]
```

## Create standalone exam session

This call creates a "standalone" session; that is, it registers a new student and grants temporary access to a single exam. It returns a unique URL used to start that exam, and an exam session ID that can be used to track exam progress.

**/exam.create\_standalone\_session**

The call has the following parameters:

Parameter	Description	Required	Default

Parameter	Description	Required	Default
exam_id	A number that identifies an exam.	Yes	
student_name	The name of the student.	No	
student_email	The email of the student. If this is specified, the student will receive an email which includes the link to start the exam.	No	
access_ip_address	If this value is specified, access to the exam is restricted to this IP address.	No	
access_duration	The duration of access to the exam. The value is specified in minutes.	No	60
exit_url	A URL that the student will be redirected to when the exam is completed. If specified, this session-specific URL overrides any fixed exit URL configured for the exam.	No	

The call returns the following:

Property	Description
person_id	A number that uniquely identifies the person (student).
exam_session_id	A number that uniquely identifies the exam session.
exam_session_url	A unique link (URL) that can be used to start the exam. The link is valid for a specific duration, and for a specific IP address if that was specified.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/exam.create_standalone_session -d "exam_id=1234&student_name=john%20doe"
```

Example Response:

```
1. [{
2.   "exam_session_id": "1234",
3.   "exam_session_url": "https://secure.e-learning.no/exam-start/example-company/1234/ILQ4AM1B2T9ONOD3E8MRNSAZXKJWCI50LZH5",
4.   "person_id": "5551"
5. }]
```

## Get exam results

This call retrieves the exam results for a given exam session ID.

## `/exam.get_results`

The call has the following parameters:

Parameter	Description	Required	Default
exam_session_id	A number that identifies an Exam Session.	Yes	
include_answers	A boolean that indicates if questions and answers should be included in the response.	No	N

The call returns the following:

Property	Description
exam_session_status	A string that indicates the status of the exam session. Possible values are NOT_STARTED , IN_PROGRESS or COMPLETED .
exam_session_result	A string that indicates the result of the exam session. Possible values are UNDEFINED (if the session has not been completed), PASSED or FAILED .
exam_session_score	The student's exam score, expressed as a percentage of correct answers.
list	A list of questions and answers for the exam session.

Example Request:

```
1. curl https://secure.e-learning.no/web/services/exam.get_results -d "exam_session_id=1234"
```

Example Response:

```
1. [{
2.   "exam_session_status": "COMPLETED",
3.   "exam_session_result": "FAILED",
4.   "exam_session_score": "60"
5. }]
```

