_Basic Usage Examples

Using Browsers Development Tools

Best way to get examples is to use the browsers development tools to capture the browsers network communication while using translate5 through the frontend.

All below listed curl examples are generated via Google Chromes DevTools.

Authentication (POST example)

First of all we have to authenticate us at the translate5 installation.The authentication is session based.

To authenticate we POST the data parameter containing the JSON data to the /editor/session URL.

More information about the session API.

Curl Example

```
curl 'https://demo.translate5.net/editor/session' -X POST -H 'Accept: application/json' \
     -H 'Content-Type: application/x-www-form-urlencoded; charset=UTF-8' \
     --data-urlencode 'data={"login":"manager","passwd":"asdfasdf"}'
```

On success this results in a similar JSON as the following. All further calls need the returned sessionId then.

JSON Result

```
{"sessionId":"HERE_WILL_BE_YOUR_SESSION_ID_THEN","sessionToken":"HERE_WILL_BE_YOUR_SESSION_TOKEN_THEN"}
```


The examples can be tested with the commandline curl. In addition there are some online tools where you can directly paste the curl examples and test them too, for example https://reqbin.com/
Unfortunately the parameter --data-urlencode used in the examples is not usable in the online tools, so the data must be encoded first and send as raw data. So the line with --data-urlencode of the above example would change to:

--data-raw 'data=%7B%22login%22%3A%22manager%22%2C%22passwd%22%3A%22asdfasdf%22%7D'

List translation tasks (GET examples)

As next step we can list all tasks available in translate5.

More information about the task API.

Curl Example

```
curl 'https://demo.translate5.net/editor/task?start=0&limit=20' -X GET -H 'Accept: application/json' -H
'Cookie: zfExtended=HERE_WILL_BE_YOUR_SESSION_ID_THEN'
```

On success this results in a JSON containing the first (start=0) 20 (limit=20) tasks in translate5.

To get a single task you need the task id (from the above returned JSON), then you can access the single task (replace TASKID accordingly):

Curl Example

```
curl 'https://demo.translate5.net/editor/task/TASKID' -X GET -H 'Accept: application/json' -H 'Cookie:
zfExtended=HERE_WILL_BE_YOUR_SESSION_ID_THEN'
```

On success this results in a JSON containing the requested task.

Filtering data

See Filtering, sorting and paging and other GET parameters.

Change Taskname (PUT example)

We change the taskname just to demonstrate a PUT request to change a data entity on the server.

More information about the task API.

To update the taskname the TASKID is needed again, also the entityVersion of the task. The entityVersion is contained in the previously fetched task entity. It is needed to prevent that two different users change an entity at the same time.

The entityVersion is not needed for all entities, only if specified in the API.

Curl Example

```
curl 'https://demo.translate5.net/editor/task/TASKID' -X PUT -H 'Cookie:
zfExtended=HERE_WILL_BE_YOUR_SESSION_ID_THEN' \
     -H 'Content-Type: application/x-www-form-urlencoded; charset=UTF-8' -H 'Accept: application/json' \
     --data-urlencode 'data={"taskName":"New Task Name","entityVersion":123}'
```

On success this results in a JSON containing the changed task.

Import Task (POST example with fileupload)

Due historical reasons coming from the used frontend framework, POST requests containing file uploads are different as the above shown POSTs without files.

More information about the task API.

A POST with fileupload contains the entities attributes not as JSON, but as plain form fields. This a known caveat.

Please change also the XLF file path to the file which should be uploaded.

Curl Example

```
curl 'https://demo.translate5.net/editor/task' -X POST  \
  -H 'Cookie: zfExtended=HERE_WILL_BE_YOUR_SESSION_ID_THEN' \
  -F "format=jsontext" \
  -F "taskName=This is a import test" \
  -F "sourceLang=de" \
  -F "targetLang=it" \
  -F "edit100PercentMatch=1" \
  -F "lockLocked=1" \
  -F "importUpload=@/PATH/TO/YOUR/FILE.xlf"
```

On success this results in a JSON containing the importing task.

Since a import is running in the background, the so created task will be returned with a state = import. For checking if the import is done, a callback should be configured: task import push callback

Associate users to a task

With the following example users can be added to a task to perform jobs on it.

More information about the task user association API.

To add such an association the task and user GUIDs are needed.

Curl Example

```
curl 'https://demo.translate5.net/editor/taskuserassoc/' \
    -X POST -H 'Accept: application/json' \
    -H 'Cookie: zfExtended=HERE_WILL_BE_YOUR_SESSION_ID_THEN' \
    --data-urlencode 'data={"taskGuid":"TASKGUID","userGuid":"USERGUID","role":"translator","state":"open",
"entityVersion":TASKVERSION}'
```

TASKGUID and USERGUID must be replaced with desired the values. TASKVERSION must be the current value of the tasks entityVersion. This is to ensure that no one has modified the task in the meantime.

For all possible roles and state values see the workflow description.

Export Task

With the following example the task will exported and saved as export.zip to the local disk.

More information about the task API.

To export the task the TASKID is needed.

Curl Example

```
curl 'https://demo.translate5.net/editor/task/export/id/TASKID' -X GET -H 'Accept: application/json' -H
'Cookie: zfExtended=HERE_WILL_BE_YOUR_SESSION_ID_THEN' --output export.zip
```

On success this results in a file "export.zip" containing the task data.

Delete Task (DELETE example)

We DELETE the task again just to demonstrate a DELETE request.

More information about the task API.

To delete the task the TASKID is needed. Also the entityVersion of the task is needed. The entityVersion is contained in the previously fetched task entity. It is needed to prevent that a user deletes an entity, which was modified on the server side in the mean time. In a DELETE request the entityVersion must be given as a HTML header field "Mqi-Entity-Version".

The entityVersion is not needed for all entities, only if specified in the API.

Curl Example

```
curl 'https://demo.translate5.net/editor/task/TASKID' -X DELETE -H 'Cookie:
zfExtended=HERE_WILL_BE_YOUR_SESSION_ID_THEN' \
    -H 'Accept: application/json' -H 'Mqi-Entity-Version: 123'
```

On success this results in a JSON containing the deleted task.

Trigger manually a workflow action for a task

In special cases some workflow actions have to been triggered manually for a task.

For Details and Configuration see the Notes about Workflow Actions.

Calling the below request via POST will trigger the notifyAllUsers notification which sends an E-Mail to all associated reviewers.

For the request the TASKID is needed, and the authenticated API user have at least the role "pm".

```
curl 'https://demo.translate5.net/editor/task/TASKID/workflow' -X POST -H 'Cookie:
zfExtended=HERE_WILL_BE_YOUR_SESSION_ID_THEN' \
    -H 'Content-Type: application/x-www-form-urlencoded; charset=UTF-8' \
    -H 'Accept: application/json' --data 'trigger=notifyAllUsersAboutTaskAssociation' --compressed
```