Internal REST API

Note that all these APIs are private and do NOT come with a deprecation policy. Although most of them have been stable for years, Blackboard reserves to change these APIs at will. The new V2 API will come with stronger stability guarantees.

Authentication

OAuth 1.0 Authentication

Ally has support for "simple" OAuth 1.0 authentication by re-purposing the LTI credentials. It does rely on the caller telling us the role of the user who is interacting with the API. Therefore it is necessary to add the following parameters:

- *userId:* The internal LMS user id of the user making the call to the Ally REST API
- role: The role of the user. This can one be of:
    - *student*: In case the calling user is a student
    - *course-manager*: In case the calling user is an instructor/teacher or somebody with edit permissions in a course
    - *administrator*: In case the user is somebody can see all data across the LMS
- *courseId:* The id of the course. This is not required when the user is an administrator

Example NodeJS script to download the zipped up CSV reports (assuming you have *request* installed):

```
const fs = require('fs');
const request = require('request');

const options = {
    'method': 'GET',
    'oauth': {
        'consumer_key': '<your lti key>',
        'consumer_secret': '<your lti secret>'
    },
    'url': 'https://prod.ally.ac/api/v1/<your client id>/reports/csv?role=administrator&userId=1'
};
request(options).pipe(fs.createWriteStream('ally.zip'));
```

JWT Authentication

This is only applicable to Canvas clients

Ally's REST APIs can be authenticated through JWT. It takes only a client id and an LTI secret to authenticate to Ally. The specific body of the JWT payload in the internal APIs is dependent on the type of LMS that the client is. The following is an example of generating a JWT token that can be used for a Moodle LMS client:

For purposes of this example, you would need to install `jwtgen`, however any utility to generate a JWT token will work:

```
$ npm install -g jwtgen
```

Then, to generate a token, using the LTI secret (the `-s` option), make the following claims (`-c`):

For **Canvas** LMSes:

```
$ jwtgen -a HS256 -s "qyAUWonHn1fW99PxBrTQFtBFObLWDxB" \
    -c "roles=urn:lti:role:ims/lis/Administrator" \
    -c 'custom_canvas_course_id="519"' \
    -c 'custom_canvas_user_id="4"' \
    -c "launch_presentation_locale=en" \
    -c "launch_presentation_return_url=https://ally.instructure.com" \
    -e 3600 -v
algorithm: HS256

claims:
{
  "iat": 1546599689,
  "exp": 1546603289,
  "roles": "urn:lti:role:ims/lis/Administrator",
  "custom_canvas_course_id": "519",
  "custom_canvas_user_id": "4",
  "launch_presentation_locale": "en",
  "launch_presentation_return_url": "https://ally.instructure.com"
}

headers:
{
  "typ": "JWT",
  "alg": "HS256"
}

token:
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.
eyJpYXQiOjE1NDY1OTk2ODksImV4cCI6MTU0NjYwMzI4OSwicm9sZXMiOiJ1cm46bHRpOnJvbGU6aW1zL2xpcy9BZG1pbmlzdHJhdG9yIiwiY3VzdG9tX2NhbnZhc19jb3Vyc2VfaWQiOiI1MTkiLCJjdXN0b21fY2FudmFzX3VzZXJfaWQiOiI0IiwibGF1bmNoX3ByZXNlbnRhdGlvbl9sb2NhbGUiOiJlbiIsImxhdW5jaF9wcmVzZW50YXRpb25fcmV0dXJuX3VybCI6Imh0dHBzOi8vYWxseS5pbnN0cnVjdHVyZS5jb20ifQ.
4Wkxq6USTAGrTgjqPVa-DQz0g1xmU3Spw63-BcG1l8s
```

For **Wcm, Moodle and D2L** LMSes:

```
$ jwtgen -a HS256 -s "vD7FoQpPFGO4STQrfi3xH23JkCsiC3Gz" -e 3600 -v \
   -c "course_id=1010" \
   -c "roles=urn:lti:role:ims/lis/Instructor" \
   -c "user_id=7" \
   -c "locale=en" \
   -c "return_url=http://moodle.dev"
algorithm: HS256

claims:
{
  "iat": 1507569748,
  "exp": 1507573349,
  "course_id": 1010,
  "roles": "urn:lti:role:ims/lis/Instructor",
  "user_id": 7,
  "locale": "en",
  "return_url": "http://moodle.dev"
}

headers:
{
  "typ": "JWT",
  "alg": "HS256"
}

token:
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.
eyJpYXQiOjE1MDc1Njk3NDgsImV4cCI6MTUwNzU3MzM0OSwiY291cnNlX2lkIjoxMDEwLCJyb2xlcyI6InVybjpsdGk6cm9sZTppbXMvbGlzL0luc3RydWN0b3IiLCJ1c2VyX2lkIjo3LCJsb2NhbGUiOiJlbiIsInJldHVybl91cmwiOiJodHRwOi8vbW9vZGxlLmRldiJ9.R6qgSGcpkMCueMl-
Ip5Pe-RPoZ_hRdj5dQNV0_1Uzio
```

For **Learn** LMSes the claims are:

- userId (string)
- courseId (string)
- courseRole (string – "privileged" gives you course manager rights, anything else grants student privileges)
- fileIds (array of strings)

The JWT token (the `token:` at the bottom can be copied and verified at http://jwt.io).

This token should now be used as the `Authorization: Bearer <token>` token when issuing requests for information from Ally to the below APIs. Note that the **roles=** parameter should be a comma-separated list of LTI roles that the user holds in the course context.

Note that some contexts for issuing requests are not in a course context (e.g., requesting institutional report information). In this case, the `course_id` parameter does not matter, if `urn:lti:role:ims/lis/Administrator` is found in the roles list, then the request will be authenticated as it is a global role.

Endpoints

Reports

`GET /api/v1/:clientId/reports/institution?groupBy=(month|year|term)`

Get the institution accessibility report data for the client identified by `:clientId`. Can be requested aggregated by calendar `month`, calendar academic `year`, or `term`.

Requires role: urn:lti:role:ims/lis/Administrator

Example:

```
$ curl -H"Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.
eyJpYXQiOjE1MDc2MzQ5NjQsImV4cCI6MTUwNzYzODU2NCwiY291cnNlX2lkIjoxMDEwLCJyb2xlcyI6InVybjpsdGk6cm9sZTppbXMvbGlzL0Fk
bWluaXN0cmF0b3IiLCJ1c2VyX2lkIjo3LCJsb2NhbGUiOiJlbiIsInJldHVybl91cmwiOiJodHRwOi8vbW9vZGxlLmRldiJ9.
Z5QFdgHe7lxZFNwrZ8R3jkxi_8aH3qSCuf0qG3vxZCE" "https://performance.ally.ac/api/v1/51/reports/institution?
groupBy=term" | underscore print
[
  {
    "future": false,
    "score": { "before": 0.2008850058402814, "after": 0.25084829888804644, "aafs": 0.28021604056320293 },
    "total": 280,
    "files": { "image": 169, "document": 14, "pdf": 38, "unknown": 56, "presentation": 3 },
    "courses": 11,
    "id": "",
    "issues": {
      "ImageSeizure": { "count": 11, "appliesTo": 169 },
      "HeadingsSequential": { "count": 1, "appliesTo": 55 },
      "Scanned": { "count": 5, "appliesTo": 55 },
      "Title": { "count": 10, "appliesTo": 55 },
      "TableHeaders": { "count": 8, "appliesTo": 55 },
      "LanguagePresence": { "count": 32, "appliesTo": 55 },
      "HeadingsPresence": { "count": 8, "appliesTo": 55 },
      "ImageOcr": { "count": 62, "appliesTo": 169 },
      "Parsable": { "count": 3, "appliesTo": 38 },
      "ImageContrast": { "count": 35, "appliesTo": 169 },
      "ImageDecorative": { "count": 168, "appliesTo": 169 },
      "AlternativeText": { "count": 7, "appliesTo": 55 },
      "ImageDescription": { "count": 166, "appliesTo": 169 },
      "Contrast": { "count": 20, "appliesTo": 55 },
      "Tagged": { "count": 20, "appliesTo": 38 },
      "LibraryReference": { "count": 55, "appliesTo": 55 }
    },
    "title": ""
  }
]
```

`GET /api/v1/:clientId/reports/courses/:courseId`

Get the course accessibility report.

Requires role: urn:lti:role:ims/lis/Administrator **or** urn:lti:role:ims/lis/Instructor with `course_id=`**`:courseId`**

Example:

```
$ curl -H"Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.
eyJpYXQiOjE1MDc1Njk3NDgsImV4cCI6MTUwNzU3MzM0OSwiY291cnNlX2lkIjoxMDEwLCJyb2xlcyI6InVybjpsdGk6cm9sZTppbXMvbGlzL0luc3RydWN0b3IiLCJ1c2VyX2lkIjo3LCJsb2NhbGUiOiJlbiIsInJldHVybl91cmwiOiJodHRwOi8vbW9vZGxlLmRldiJ9.R6qgSGcpkMCueMl-
Ip5Pe-RPoZ_hRdj5dQNV0_1Uzio" "https://performance.ally.ac/api/v1/51/reports/courses/1010" | underscore print
{
  "name": "Chemistry 101",
  "url": "http://allyclone-sandbox.mrooms.net/course/view.php?id=1010",
  "students": 2,
  "score": { "before": 0.16612737141556672, "after": 0.5467933796762355, "aafs": 0.6174205854234196 },
  "total": 16,
  "files": { "pdf": 4, "document": 8, "image": 4 },
  "code": "Chemistry",
  "id": "1010",
  "issues": {
    "HeadingsSequential": { "count": 1, "appliesTo": 12 },
    "Title": { "count": 1, "appliesTo": 12 },
    "TableHeaders": { "count": 2, "appliesTo": 12 },
    "LanguagePresence": { "count": 10, "appliesTo": 12 },
    "HeadingsPresence": { "count": 3, "appliesTo": 12 },
    "ImageDecorative": { "count": 4, "appliesTo": 4 },
    "AlternativeText": { "count": 2, "appliesTo": 12 },
    "ImageDescription": { "count": 3, "appliesTo": 4 },
    "Contrast": { "count": 9, "appliesTo": 12 },
    "Tagged": { "count": 2, "appliesTo": 4 },
    "LibraryReference": { "count": 12, "appliesTo": 12 }
  }
}
```

`GET /api/v1/:clientId/reports/courses/:courseId/files?fileIds=:fileId1,:fileId2,...`

Get the file report information for a set of files in a course.

Depending on the role of the user, more or less information will be returned.

Requires role: urn:lti:role:ims /lis/Administrator **or** urn:lti:role:ims/lis/Instructor with `course_id=`**`:courseId or`** `urn:lti:role:ims/lis/Student with course_id=:`**`courseId`**

Example:

```
// Student
$ curl -H"Authorization: OAuth ...." "http://ally.local/api/v1/1/reports/courses/1/files?fileIds=2180,
2181&role=student&courseId=1" | underscore print
{
    "2008": {
        "name": "Personalizing Web Search using Long Term Browsing History.pdf",
        "size": 833041,
        "decorative": null,
        "description": null,
        "seizureRisk": false,
        "creator": "Mac OS X 10.11.2 Quartz PDFContext",
        "id": "2008",
        "availableAlternativeFormats": [
            "Braille",
            "Epub",
            "Html"
        ],
        "isVersioned": false,
        "type": "pdf",
        "libraryReference": null,
        "producer": "TeX"
```

```
    },
    "2180": {
        "name": "2.1 Write a Research Method Section Assignment Final.html",
        "size": 5408,
        "decorative": null,
        "description": null,
        "seizureRisk": false,
        "creator": null,
        "id": "2180",
        "availableAlternativeFormats": [
            "Tts",
            "Epub",
            "Braille"
        ],
        "isVersioned": false,
        "type": "html-page",
        "libraryReference": null,
        "producer": null
    }
}


// Course manager (i.e. an instructor) or administrator
// Note how there is now score information available
{
    "2008": {
        "name": "Personalizing Web Search using Long Term Browsing History.pdf",
        "size": 833041,
        "decorative": null,
        "description": null,
        "score": 0.05,
        "seizureRisk": false,
        "suggestions": {
            "Title": 0.06899917335595727,
            "LanguagePresence": 0.09749793338989321,
            "Contrast": 0.05,
            "Tagged": 0.9334615610520135,
            "LibraryReference": 1
        },
        "creator": "Mac OS X 10.11.2 Quartz PDFContext",
        "id": "2008",
        "results": {
            "Title": 0,
            "LanguagePresence": 0,
            "Contrast": 0.999956492418804,
            "Tagged": 0,
            "LibraryReference": 0
        },
        "availableAlternativeFormats": [
            "Braille",
            "Epub",
            "Html"
        ],
        "isVersioned": false,
        "type": "pdf",
        "libraryReference": null,
        "producer": "TeX"
    },
    "2180": {
        "name": "2.1 Write a Research Method Section Assignment Final.html",
        "size": 5408,
        "decorative": null,
        "description": null,
        "score": 1,
        "seizureRisk": false,
        "suggestions": {},
        "creator": null,
        "id": "2180",
        "results": {},
        "availableAlternativeFormats": [
            "Tts",
```

```
            "Epub",
            "Braille"
        ],
        "isVersioned": false,
        "type": "html-page",
        "libraryReference": null,
        "producer": null
    }
}
```

`POST /api/v1/:clientId/reports/courses/:courseId/files`

Similar to the above endpoint, but allows for retrieving many files at a time as query strings are limited to a certain number of characters.

Ensure the *Content-Type* header is set to *application/*json and submit a valid JSON array containing objects that hold the ids of the files you would like to submit.

Depending on the role of the user, more or less information will be returned. The returned array will be in the same order as the provided JSON array. If Ally does not hold any information for a given file id, a *null* value will be returned.

Requires role: urn:lti:role:ims /lis/Administrator **or** urn:lti:role:ims/lis/Instructor with `course_id=`**`:courseId or`** `urn:lti:role:ims/lis/Student with course_id=:`**`courseId`**

Example:

```
// Student
$ curl -H"Authorization: OAuth ...." -H "Content-Type: application/json " -X POST -d '[{"id":"2180"}, {"id":"
2008"}]'http://ally.local/api/v1/1/reports/courses/1/files?role=student&courseId=1" | underscore print
[
    {
        "name": "2.1 Write a Research Method Section Assignment Final.html",
        "size": 5408,
        "decorative": null,
        "description": null,
        "seizureRisk": false,
        "creator": null,
        "id": "2180",
        "availableAlternativeFormats": [
            "Tts",
            "Epub",
            "Braille"
        ],
        "isVersioned": false,
        "type": "html-page",
        "libraryReference": null,
        "producer": null
    },
    {
        "name": "Personalizing Web Search using Long Term Browsing History.pdf",
        "size": 833041,
        "decorative": null,
        "description": null,
        "seizureRisk": false,
        "creator": "Mac OS X 10.11.2 Quartz PDFContext",
        "id": "2008",
        "availableAlternativeFormats": [
            "Braille",
            "Epub",
            "Html"
        ],
        "isVersioned": false,
        "type": "pdf",
        "libraryReference": null,
        "producer": "TeX"
    }
]


// Course manager (i.e. an instructor)
```

```
// Note how there is now score information available
[
    {
        "name": "2.1 Write a Research Method Section Assignment Final.html",
        "size": 5408,
        "decorative": null,
        "description": null,
        "score": 1,
        "seizureRisk": false,
        "suggestions": {},
        "creator": null,
        "id": "2180",
        "results": {},
        "availableAlternativeFormats": [
            "Tts",
            "Epub",
            "Braille"
        ],
        "isVersioned": false,
        "type": "html-page",
        "libraryReference": null,
        "producer": null
    },
    {
        "name": "Personalizing Web Search using Long Term Browsing History.pdf",
        "size": 833041,
        "decorative": null,
        "description": null,
        "score": 0.05,
        "seizureRisk": false,
        "suggestions": {
            "Title": 0.06899917335595727,
            "LanguagePresence": 0.09749793338989321,
            "Contrast": 0.05,
            "Tagged": 0.9334615610520135,
            "LibraryReference": 1
        },
        "creator": "Mac OS X 10.11.2 Quartz PDFContext",
        "id": "2008",
        "results": {
            "Title": 0,
            "LanguagePresence": 0,
            "Contrast": 0.999956492418804,
            "Tagged": 0,
            "LibraryReference": 0
        },
        "availableAlternativeFormats": [
            "Braille",
            "Epub",
            "Html"
        ],
        "isVersioned": false,
        "type": "pdf",
        "libraryReference": null,
        "producer": "TeX"
    }
]
```

CSV export

Both CSV endpoints below work asynchronously, a background process is spawned to generate CSV export. The logic is the following:

- If the CSV export already exists, a download URL will be returned.
- If it does not, a JSON object with a process id will be returned.

The important thing is to provide the token query parameter with a unique value. The unique value is part of the resulting filename of CSV export which is stored and CACHED on Ally servers. If the request is called with the same token more than once and the CSV export already exists, that already generated snapshot will be returned. If new or up to date CSV export is needed, just provide different token to the request and wait for a process to complete. Convenient token value could be a timestamp/date etc.

Example:

```
https://prod.ally.ac/api/v1/15/reports/csv?department=1&token=2022-06-15-14-46

# Response if the CSV export still needs to be generated, you can use the "GET /api/v1/processes/:processId"
described below to find out whether the process completed
{
    "processId": "4d5f7611-dcca-4f14-824b-2a94ef015b25",
    "status": "Pending"
}

# After the process completes and CSV export is generated
{
    "url": "https://ally-production.s3.amazonaws.com/csv/15//ally-Fronteer_Canvas_Instance-Blackboard_Ally-2022-
06-15-14-46.zip..."
}
```

`GET /api/v1/:clientId/reports/csv`

Get a zip file with CSV data for the entire institution. The CSV file will contain 4 CSV files:

1. A by year view of all the data
2. A by month view of all the data
3. A by term view of all the data
4. A by course view of all the data. Each course will have 1 entry in the CSV file.

This endpoint is not available via JWT authentication for certain LMSes.

Requires role: urn:lti:role:ims/lis/Administrator

`GET /api/v1/:clientId/reports/courses/:courseId/csv`

Get the CSV report for a course.

Requires role: urn:lti:role:ims/lis/Administrator **or** urn:lti:role:ims/lis/Instructor with `course_id=`**`:courseId`**

Alternative formats

`GET /api/v1/:clientId/formats/:courseId/:fileId/:format?acceptTOU=<true|false>[&language=<language>]`

Acceptable format values:

- Braille
- Epub
- Html
- OcredPdf
- Pdf
- Source
- Translation
- Tts

Acceptable language values:

- af
- ar
- bs
- bs-Latn
- bg
- yue
- zh-Hans-CN
- zh-Hant-TW
- ca
- hr
- cs
- da
- nl
- en
- et
- fi
- fr

- de
- el
- ht
- hi
- hu
- he
- id
- it
- ja
- ko
- lv
- lt
- ms
- mt
- no
- fa
- pl
- pt
- ro
- ru
- es
- sr-Cyrl
- sr-Latn
- sk
- sl
- sw
- sv
- th
- tr
- uk
- ur
- vi
- cy

Get an alternative format for a file. If the alternative format already exists, a download URL will be returned. If it does not, a JSON object with a process id will be returned

Requires role: urn:lti:role:ims/lis/Learner (or higher)

Example:

```
$ curl -H"Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.
eyJpYXQiOjE1MDc2MzQ5NjQsImV4cCI6MTUwNzYzODU2NCwiY291cnNlX2lkIjoxMDEwLCJyb2xlcyI6InVybjpsdGk6cm9sZTppbXMvbGlzL0Fk
bWluaXN0cmF0b3IiLCJ1c2VyX2lkIjo3LCJsb2NhbGUiOiJlbiIsInJldHVybl91cmwiOiJodHRwOi8vbW9vZGxlLmRldiJ9.
Z5QFdgHe7lxZFNwrZ8R3jkxi_8aH3qSCuf0qG3vxZCE" "https://performance.ally.ac/api/v1/51/formats/9/Html?
acceptTOU=true" | underscore print

# If the alternative format already exists:
{
    "url": "https://ally-performance.s3.amazonaws.com/files/7c/25/4b/2d/ally_aaf_123....."
}


# If the format still needs to be generated (see endpoint below)
{
    "process_id": "acc6d2ed-0a20-4f21-b8e8-9e50d985a21c",
    "status": "Pending"
}
```

`GET /api/v1/processes/:processId`

Requires role: none

Example:

```
curl https://performance.ally.ac/api/v1/processes/acc6d2ed-0a20-4f21-b8e8-9e50d985a21c
# When the process has started, but hasn't completed yet
{
    "process_id": "acc6d2ed-0a20-4f21-b8e8-9e50d985a21c",
    "status": "InProgress"
}


# When the process has completed
{
    "process_id": "acc6d2ed-0a20-4f21-b8e8-9e50d985a21c",
    "status": "Succeeded"
}

# When the process has failed
{
    "process_id": "acc6d2ed-0a20-4f21-b8e8-9e50d985a21c",
    "status": "Failed"
}
```