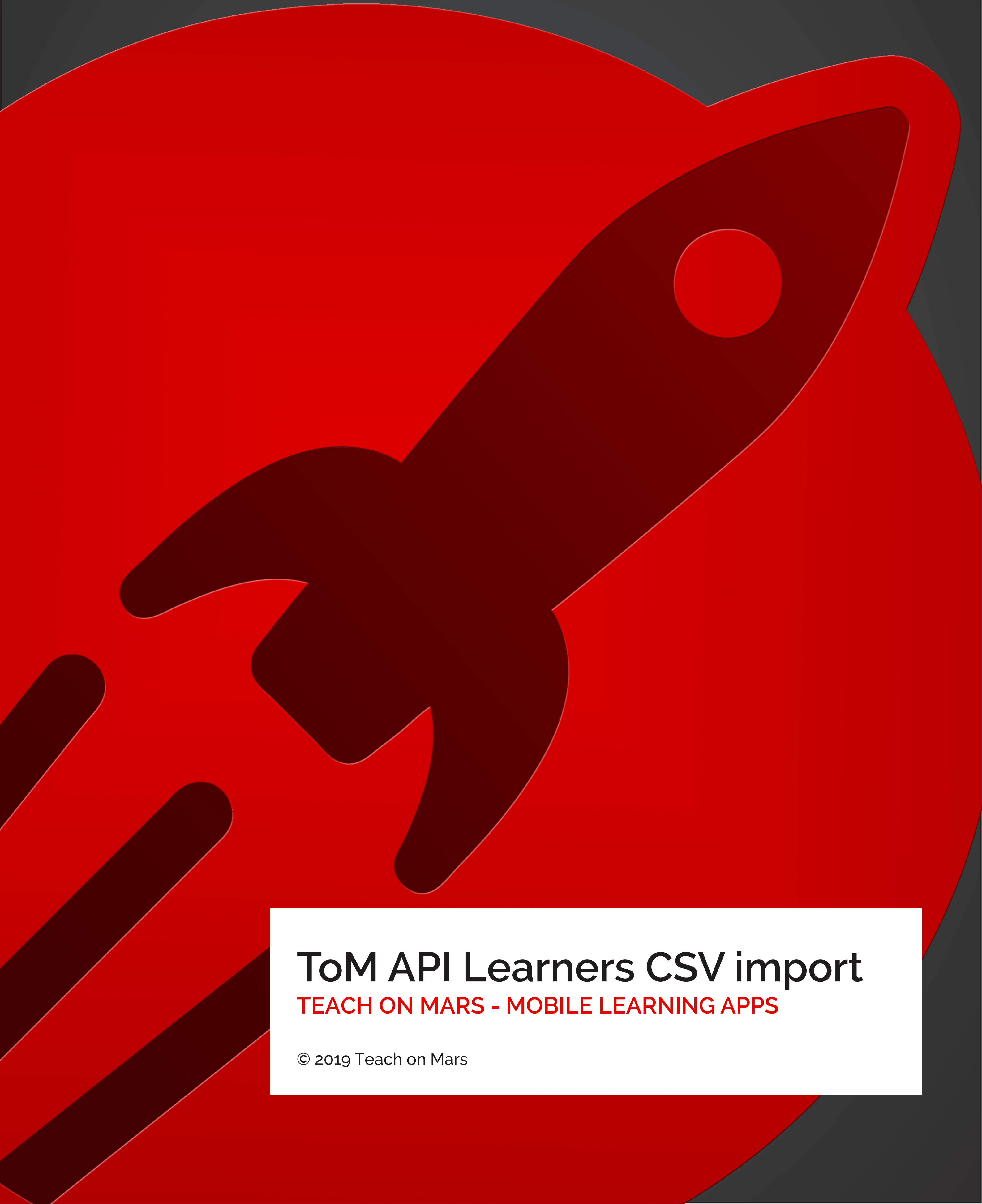
TEACH ON MARS



ToM API Learners CSV import

TEACH ON MARS - MOBILE LEARNING APPS

Introduction

When integrating Teach on Mars with a third-party application, one of the most important task is to deal with the way the learners are synchronized between the systems.

Import learners with CSV file

To update and create a massive amount of learners in the Mission Center, the best way is to send a batch of CSV data. Here's how it works.

Overview

The learners CSV import works in two steps:

- **Send the data for validation**
  - There you get a report on the data you have sent
- **Confirm the import**
  - If the first step was successful, this is where the import is actually applied to the database

Send the data

Prepare a CSV file with the following requirements:

- Delimiter: **;**
- Encoding: UTF-8
- First line (header line) defines the fields

You can obtain such a file using the learner export in the Mission Center, or with an automated export process. Here's how it should look:

```
login;firstname;lastname;email;status;metacountry(Country);metadepartment
john.doe;John;Doe;john.doe@company.com;A;"France";Sales
```

Here's a list of the allowed fields:

Field	Description
login	The login of the learner. Used by the learner to log in the application. This field is mandatory in the data.
firstname	The firstname of the learner
lastname	The lastname of the learner
email	The email of the learner
password	This will set a new password for the learner. If you leave it empty (or omit the column) the password will not change.
status	Either A (active) or I (inactive). Inactive learners don't have access to the application.
lang	Force the language of the learner. It is useful to set the language of the emails and communications sent to the learner before he logged in the app. Enter a language code like *fr*, *zh-CN*, etc.
sandbox_tester	Either Y or N. If Y, the learner will be able to see *test versions* of the training courses in the application.
meta*	Custom fields configured in the Mission Center

Then you'll need to work on this data so it can fit into the following JSON payload. Here's what you need to do:

- Replace *new lines* with *\n* characters
- Escape " quotes so it doesn't break the JSON format

Request

```
POST /api/v3/learner/csv

{
  "data":
"login;firstname;lastname;email;status;metacountry(Country);metadepartment\njo
hn.doe;John;Doe;john.doe@company.com;A;\"France\";Sales"
}
```

Options

It is possible to add options when importing learners:

Updates only

```
POST /api/v3/learner/csv

{
  "data":
"login;firstname;lastname;email;status;metacountry(Country);metadepartment\njo
hn.doe;John;Doe;john.doe@company.com;A;\"France\";Sales",
  "options": {
    "updateOnly": true
  }
}
```

This option will ignore account creations and proceed with learner updates only

Responses

200 OK

Whats does it mean? The webservice has accepted the file, which is well formatted. But the content of the file might still contain errors.

```
{
  "meta": {
    "code": 200,
    "msg": "OK"
  },
  "response": {
    "importId": "hfdiFDIufsid865hf"
    "status": "S",
    "report": {
      "updated": 10,
      "activated": 5,
      "deleted": 0,
      "added": 40,
      "addedinactive": 0,
      "errors": [


      ]
    }
  }
}
```

What now?

importId: It will be useful to call the next service to confirm this import. (see next section)

status: Either S (successful) or E (error)

S (successful)	No errors were found in the content, a quick report of what will happen in the learners database is given to you in the *report* part of the response. If status = S, you can use the importID in the second step.
E (error)	Some errors we found in the content. A report of the errors is listed in the *report.errors* part of the response. These errors are human readable, so they are suited for a report via email or any interface/log.

400 - LEARNER_IMPORT_PARAM_DATA_MISSING

```
{
  "meta": {
    "code": 400,
    "msg": "LEARNER_IMPORT_PARAM_DATA_MISSING"
  },
  "response": {}
}
```

Whats does it mean? The webservice could not extract the *data* parameter from the request.

What might be the reason?

- The *data* key is either missing, misspelled or misplaced in the request JSON payload
- The *Content-Type: application/json* is missing from the request
- There is an error in the JSON encoding of the payload, which prevents the service from parsing it

400 - LEARNER_IMPORT_INVALID_CSV

```
{
  "meta": {
    "code": 400,
    "msg": "LEARNER_IMPORT_INVALID_CSV"
  },
  "response": {}
}
```

Whats does it mean? The webservice could read the data, but it is empty.

What might be the reason?

- The file put into the *data* param is empty

406 - LEARNER_IMPORT_NO_DATA

```
{
  "meta": {
    "code": 406,
    "msg": "LEARNER_IMPORT_NO_DATA"
  },
  "response": {}
}
```

Whats does it mean? The webservice could read the data, but it is empty. It is similar to the previous response but the test on the MC side is a bit different.

What might be the reason?

- The file put into the *data* param is empty

400 - LEARNER_IMPORT_LOGIN_MISSING

```
{
  "meta": {
    "code": 400,
    "msg": "LEARNER_IMPORT_LOGIN_MISSING"
  },
  "response": {}
}
```

Whats does it mean? The webservice could read the data, but the header line (with the fields) does not contain *login*.

What might be the reason?

- There is no column *login*
- The column *login* is misspelled or contains a space character

400 - LEARNER_IMPORT_ROW_TOO_MANY_VALUES

```
{
  "meta": {
    "code": 400,
    "msg": "LEARNER_IMPORT_ROW_TOO_MANY_VALUES"
  },
  "response": {}
}
```

Whats does it mean? The webservice could read the data, but some lines contain more columns than the header line.

What might be the reason?

- There must be an extra **;** in one of the lines

400 - LEARNER_IMPORT_ROW_MISSING_VALUES

```
{
  "meta": {
    "code": 400,
    "msg": "LEARNER_IMPORT_ROW_MISSING_VALUES"
  },
  "response": {}
}
```

Whats does it mean? The webservice could read the data, but some lines contain less columns than the header line.

What might be the reason?

- There must be an extra **;** in the header line
- Or a missing **;** in the data lines

400 - LEARNER_IMPORT_UNNAMED_COLUMN

```
{
  "meta": {
    "code": 400,
    "msg": "LEARNER_IMPORT_UNNAMED_COLUMN"
  },
  "response": {}
}
```

Whats does it mean? The webservice could read the data, but the header line contains some empty fields

What might be the reason?

- One field is missing in the header column (two consecutive ;)

```
{
  "meta": {
    "code": 200,
    "msg": "OK"
  },
  "response": {
    "importId": "hfdiFDIufsid865hf"
    "status": "C",
    "report": {
      "updated": 10,
      "activated": 5,
      "deleted": 0,
      "added": 40,
      "addedinactive": 0,
      "errors": [


      ]
    }
  }
}
```

```
{
  "meta": {
    "code": 400,
    "msg": "LEARNER_IMPORT_FIELD_NOT_ALLOWED"
  },
  "response": {}
}
```

400 - LEARNER_IMPORT_FIELD_NOT_ALLOWED

Whats does it mean? The webservice could read the data, but the header line contains some fields that are not allowed

What might be the reason?

- One field is not recognized by the system. Please check the list of allowed fields in the service documentation above.

Step 2: Confirm the import

Request

To confirm the import, just send a POST request with no payload to the following URL.

```
POST /api/v3/learner/csv/{importId}
```

If the importId does not correspond to an import with the status S, you will receive an error.

Options

It is possible to add options when importing learners:

Notify new learners

```
POST /api/v3/learner/csv/{importId}

{
  "options": {
    "notifyNewLearners": true
  }
}
```

This option will send an email to newly created learners if they have an email address. This email is configurable for the Mission Center, but it needs to be done through the Teach on Mars team.

Responses

<u>200 - OK</u>

Whats does it mean? The request was successful, the import has been applied. You can notice the status = C (Confirmed)

404 - Import not found

```
{
  "meta": {
    "code": 404,
    "msg": "Import not found"
  },
  "response": {}
}
```

Whats does it mean? The import was not found in the database
What might be the reason?

- The importId given to the request is wrong

400 - LEARNER_IMPORT_UNSUCCESSFUL

```
{
  "meta": {
    "code": 400,
    "msg": "LEARNER_IMPORT_UNSUCCESSFUL"
  },
  "response": {}
}
```

Whats does it mean? The import was found but does not have the status **S**

What might be the reason?

- The import related to this ID had some errors.
- The import was already confirmed. Step 2 was already done, and the same ID has been used several times.