API Documentation – Agritask

Introduction.

It's possible that a third-party software integrates with Agritask where it's possible to consume Agritask system data, update or create records.
To update or create records in Agritask system it's used a Rest API type POST where it's sent a CSV file with a pre-defined structure depending on each operation.

The POST calls must follow the structure described below:

URL: https://ricetec.agritask.com/r/api/private/imports/v4

Headers:
`WEB-SERVICE-KEY: 3a6d314d830404607d599ce76d2eb77b`

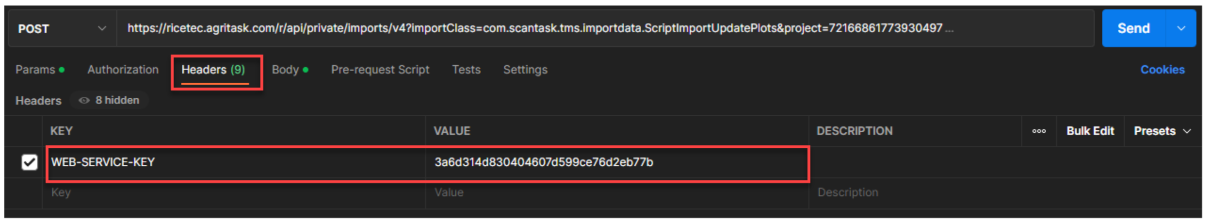
Query Params:

`importClass: com.scantask.tms.importdata.ScriptImportUpdatePlots`

`project: 72166861773930497`

Query Params:
`file`: CSV file to be send

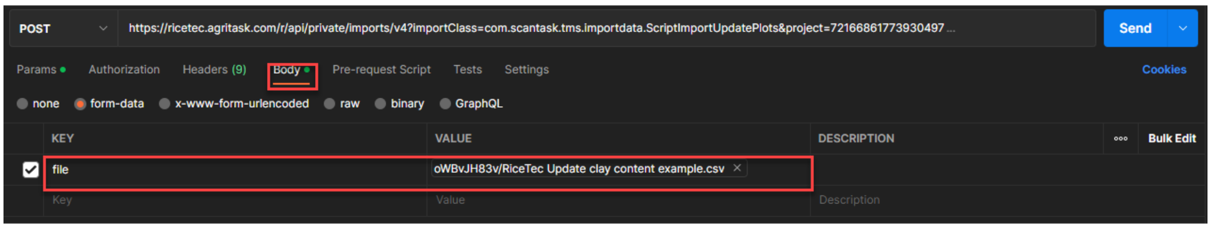
Check the example from Postman:


POST https://ricetec.agritask.com/r/api/private/imports/v4?importClass=com.scantask.tms.importdata.ScriptImportUpdatePlots&project=72166861773930497 ...

Params ● | Authorization | Headers (9) | Body ● | Pre-request Script | Tests | Settings

Headers 8 hidden

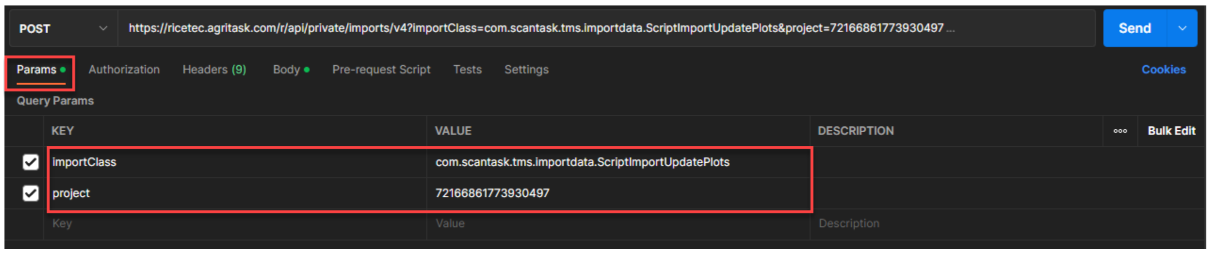
KEY	VALUE	DESCRIPTION
✓ WEB-SERVICE-KEY	3a6d314d830404607d599ce76d2eb77b	
Key	Value	Description


POST https://ricetec.agritask.com/r/api/private/imports/v4?importClass=com.scantask.tms.importdata.ScriptImportUpdatePlots&project=72166861773930497 ...

Params ● | Authorization | Headers (9) | Body ● | Pre-request Script | Tests | Settings

none [✓ form-data] x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION
✓ file	oWBvJH83v/RiceTec Update clay content example.csv	
Key	Value	Description


POST https://ricetec.agritask.com/r/api/private/imports/v4?importClass=com.scantask.tms.importdata.ScriptImportUpdatePlots&project=72166861773930497 ...

Params ● | Authorization | Headers (9) | Body ● | Pre-request Script | Tests | Settings

Query Params

KEY	VALUE	DESCRIPTION
✓ importClass	com.scantask.tms.importdata.ScriptImportUpdatePlots	
✓ project	72166861773930497	
Key	Value	Description

CSV file structure.

The CSV file must contain at least 12 columns. The number of columns can be more than that though, depending on If there are additional fields to be updated. In all cases, it's important to make sure that the columns are separated by comma and not by semi-column or point. The column names do not matter until the 13° column ([M]) forward (14° [N], 15° [O], …). From this point the column names need to be precisely the column names of the additional properties.

The columns that mandatory request values are the first and second columns ([A] and [B] respectively). These must contain the codes* of the Grower and Plot entities. These are mandatory so the system can identify which entity we are trying to update.

For our purposes, let's check below the structure of the CSV file we must send:

[A]	B	C	D	E	F	G	H	I	J	K	L	M	clay_content_percentage
FR11Q3S2Z	AL598077Z												{{float value}}

Column names:

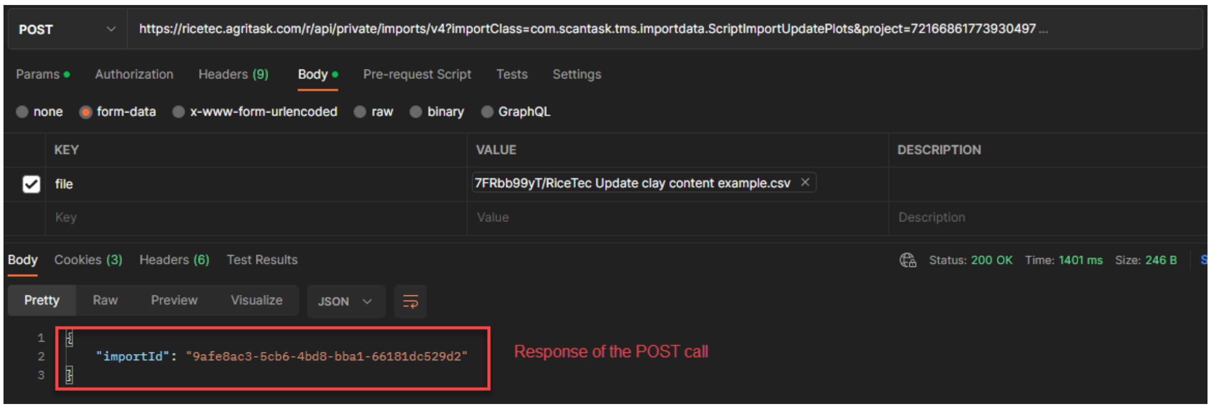
[A] grower code,
[B] plot code.
The rest of the columns are optional:
[C] date of when the plot was active (today if empty) - used to distinguish between similarly named plots or between seasons,
[D] new date of when the plot was created,
[E] new date of when the plot was closed,
[F] new name,
[G] new localized name,
[H] new second localized name,
[I] new plot area in square meters,
[J] new latitude,
[K] new longitude,
[L] new plot polygon in WKT format,
[M] new external ID,
[N] clay_content_percentage -> The column name must have exactly this name in yellow.

Example of file that can be sent to update a plot with the clay content:
The field name that the file updates is "*Test plot 1*" under the grower "*Test Grower*"

RiceTec Update clay content example.csv

Responses.

Every POST call will return a related `importId`. This `importId` can be used in a GET call to check if there were any errors in the POST request. Check the structure below:


POST https://ricetec.agritask.com/r/api/private/imports/v4?importClass=com.scantask.tms.importdata.ScriptImportUpdatePlots&project=72166861773930497 ...

	KEY	VALUE	DESCRIPTION
✓	file	7FRbb99yT/RiceTec Update clay content example.csv	
	Key	Value	Description

Status: 200 OK Time: 1401 ms Size: 246 B

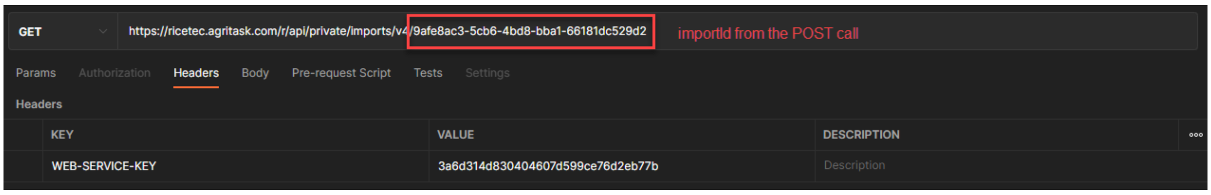
```
{
    "importId": "9afe8ac3-5cb6-4bd8-bba1-66181dc529d2"
}
```

Response of the POST call

To GET the information of the file sent, use the following structure:

URL: https://ricetec.agritask.com/r/api/private/imports/v4/
Headers: `WEB-SERVICE-KEY: 3a6d314d830404607d599ce76d2eb77b`

Add the `importId` in the request URL as below example:


GET https://ricetec.agritask.com/r/api/private/imports/v4/9afe8ac3-5cb6-4bd8-bba1-66181dc529d2

importId from the POST call

KEY	VALUE	DESCRIPTION
WEB-SERVICE-KEY	3a6d314d830404607d599ce76d2eb77b	Description

The response to the GET call follow the structure below:

```
{
    "status": "COMPLETED",
    "params": {
        "importClassName": "com.scantask.tms.importdata.ScriptImportUpdatePlots",
        "project": 72166861773930497,
        "commitChanges": true,
        "auto": false,
        "view": null,
        "timezone": null
    },
    "messages": [
        {
            "type": "WARN",
            "line": 2,
            "message": "Plot 'AL598077' not found in the database"
        },
```

In this example, and error was intentionally caused to demonstrate the error messages. Under `"messages"`, the `"type"` indicates which rows have issues. The `"line"` indicates which rows in the file has issues and the `“message”` informs the issue. In this case the final character was removed from the Plot code. The correct value would be `'AL598077Z'`