

# LabCollector

*"Your Laboratory management solution"*

User's guide

## Web Service API

Version 2.0 – November 2021



# Summary

<b>1- INTRODUCTION .....</b>	<b>3</b>
<b>2- LABCOLLECTOR API .....</b>	<b>4</b>
<b>2-1. API setup.....</b>	<b>4</b>
<b>2-2. Requests .....</b>	<b>5</b>
<i>2-2-1. API Method.....</i>	<i>5</i>
<i>2-2-3. Tool.....</i>	<i>6</i>
<b>2-3. Uniform Resource Identifier (URI) .....</b>	<b>6</b>
<i>2-3-1. GET method.....</i>	<i>6</i>
<i>General .....</i>	<i>6</i>
<i>Custom parameters .....</i>	<i>6</i>
<i>Storage .....</i>	<i>9</i>
<i>Product lot .....</i>	<i>11</i>
<i>Recipe .....</i>	<i>12</i>
<i>2-3-2. POST method.....</i>	<i>14</i>
<i>2-3-3. PUT method.....</i>	<i>16</i>
<i>2-3-4. DELETE method .....</i>	<i>18</i>
<b>3- API ERROR MESSAGES .....</b>	<b>19</b>
<b>4- LABCOLLECTOR WEB SERVICE API - ANNEX .....</b>	<b>21</b>

## 1- INTRODUCTION

The LabCollector Web Service Application Programming Interface (API) allows third-party applications to interact with LabCollector's database (modules) and add-ons (ELN and LSM).

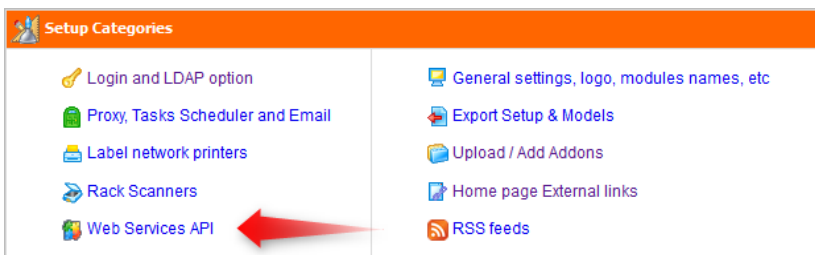
The API is based on a Representational State Transfer (REST) architecture allowing access to resources through Uniform Resource Identifier (URI) and actions on them.

**Note: Since June 2017 API v1 was discontinued and all new evolutions are in API v2.**

## 2- LABCOLLECTOR API

### 2-1. API setup

First of all, you have to declare your application in your LabCollector software. To access the application declaration setup form, log-in to LabCollector with super-administrator rights and go to **Admin > Setup** page. Then select the Web Service API link.



You are now on the Web Service API applications management page. To declare a new application, simply complete this form:

 A screenshot of the 'Web Services API' application management page. The page has a green header with a person icon and the text 'Web Services API'. Below the header, there is a blue bar with the text 'Define a new application:'. The main form area has a green background and contains the following fields: 'Name' (text input), 'Default operator' (dropdown menu with 'Select Contact' selected), and 'IP Restriction' (radio buttons for 'Yes' and 'No'). Below these fields, there is a section titled 'Modules connected to this application:' with a grid of checkboxes for various modules: Strains & Cells, Animals, test pharma, Sample ref, Plasmids, Equipment, Server LC, Primers, Chemical Structures, Parsed data, Reagents & Supplies, Documents, Analysis, Samples, Address Book, storage test, Antibodies, Microarrays, Deviation module LSM, Sequences, Clone banks, and Hirondelle. A green '+ Add' button is located on the right side of the form.

- **Name:** name of your application.
- **Modules connected to this application:** select modules in which the application can access.
- **Default operator:** select the contact that will be the default operator if you don't want to insert this information in each request.
- **IP Restriction:** security option that allows you to declare a list of IP addresses, which will be allowed to perform requests on the API.

The **Application list** shows all the applications for your LabCollector and you can, at any time, modify their scope.

You have also access to the Token which is necessary to identify your application during requests to the API.

Remote applications list:			
Application	Used Modules		IP Restriction
Name: ICollector Default operator: <input type="text" value="Select Contact"/> Token: 1eb2b363bcebdde113de4fe17f9253d94202bd9256369daf65c71e151f930bb <input type="button" value="Regenerate token"/>	<input type="checkbox"/> Strains & Cells <input type="checkbox"/> Primers <input checked="" type="checkbox"/> Samples <input type="checkbox"/> Sequences <input type="checkbox"/> Equipments <input type="checkbox"/> Documents <input type="checkbox"/> Microarrays <input type="checkbox"/> test pharma <input type="checkbox"/> Parsed data <input type="checkbox"/> storage test <input type="checkbox"/> Hirondele	<input type="checkbox"/> Plasmids <input type="checkbox"/> Reagents & Supplies <input checked="" type="checkbox"/> Antibodies <input type="checkbox"/> Animals <input type="checkbox"/> Chemical Structures <input type="checkbox"/> Address Book <input type="checkbox"/> Clone banks <input type="checkbox"/> Server LC <input type="checkbox"/> Analysis <input type="checkbox"/> Deviation module LSM <input type="checkbox"/> Sample ref	<input type="radio"/> Yes <input checked="" type="radio"/> No <input type="button" value="Update - Update app"/>

**Note:** To use this feature, you need to activate Curl on your PHP preferences.

In Linux, install the PHP-Curl package.

On windows and with our automatic installer, edit PHP.ini and uncomment extensions for Curl (extension=php\_curl.dll).

## 2-2. Requests

The dialog between third-party applications and the LabCollector web service API is based on the HTTP 1.1 protocol.

### 2-2-1. API Method

You can send HTTP or HTTPS requests to the web service with a method to act on a resource.

- **GET** method for reading a resource
- **POST** method to create a new resource
- **PUT** method to modify a resource
- **DELETE** method to delete a resource

### 2-2-2. Headers

A request to the API requires some specific HTTP/HTTPS headers:

- The **Accept** header defines the desired response format of your request, text/xml or application/json.
- The **X-LC-APP-Auth** header where you put your application token which is necessary to authorize your request to the API.
- The **X-LC-APP-Charset** header defines the character encoding of your application. It allows the API to send back the response with the appropriate

encoding and to correctly convert your POST and PUT requests to the LabCollector's character encoding (ISO 8859-1).

### 2-2-3. Tool

You can try to retrieve data from or send data to the API with some software app as Postman (<https://www.getpostman.com/>).

## 2-3. Uniform Resource Identifier (URI)

### 2-3-1. GET method

#### General

Each LabCollector module data is identified by a unique URI (see annex for a complete list of module's URI):

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]
```

This request replies the list of all data in a module.

You can do searching into module data by adding parameters to your URI. You can pass a parameter with a keyword matching a field value, like:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?name=[KEYWORD]
```

e.g.

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?name=First%20Record
```

This request returns the records where their name value contains the "First Record" keyword.

They are some custom parameters that the API uses to perform searching and filtering actions.

#### Custom parameters

- The **record\_id** parameter to specify data by its ID:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?record_id=[RECORD_ID]
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?record_id=1,19`

This request returns records with **ID 1** and **ID 19**. You can specify multiple IDs by separating them with a comma.

- The **by\_keywords** parameter performs a keyword search:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?by_keywords=[KEYWORD]
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?by_keywords=cell`

This request performs a search into all fields of all records and returns matching **cell**. You can specify multiple keywords by separating them with a comma.

- The **fields** parameters, if you want to retrieve only some fields values in the API response:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?fields=[FIELD1],[FIELD2]
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?fields=count,name`

This request returns all records from the module but with only **count** and **name** fields. You can specify multiple fields by separating them with a comma.

The request accepts now multiple values separated by a comma, for custom fields of type "select"

http://localhost/lab0/webservice/v2/samples?select\_list=68,69

GET http://localhost/lab0/webservice/v2/samples?select\_list=68,69 Send

Params Authorization Headers (2) Body Pre-request Script Tests Cookies Code

Query Params

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> select_list	68,69	
Key	Value	Description

Body Cookies (1) Headers (14) Test Results Status: 200 OK Time: 230 ms Size: 2.83 KB

Pretty Raw Preview XML

```

1 <samples>
2 <data>
56 <data>
103 </samples>

```

- The `search_on` parameter allows you to search for data. And you can use it to search by date range as follow:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]&
search_on=date_field&from=XXXXXX&to=ZZZZZ
```

If you only use FROM, the result will give all dates bigger than the FROM date. If you only use TO, it will return all the value until this date.

GET http://localhost/labcollector-5.5/webservice/index.php?v=2&module=strains&fields=count,name&search\_on=date\_updated&from=2020-01-30 10:00:00&to=2020-09-09

Params Authorization Headers (13) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> v	2	
<input checked="" type="checkbox"/> module	strains	
<input checked="" type="checkbox"/> fields	count,name	
<input checked="" type="checkbox"/> search_on	date_updated	
<input checked="" type="checkbox"/> from	2020-01-30 10:00:00	
<input checked="" type="checkbox"/> to	2020/09/09	
Key	Value	Description

Body Cookies (1) Headers (13) Test Results Status: 200 OK Time: 268 ms

Pretty Raw Preview Visualize XML

```

1 <strains>
2 <data>
3 <count>3</count>
4 <name>strain1</name>
5 </data>
6 <data>
7 <count>3</count>
8 <name>strain4</name>
9 </data>

```

- The `sort_by` parameter allows you to sort your search:



```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?sort_by=[FIELD1]_DESC
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?sort_by=name_DESC`

This request returns all records sorted in descending order on the `name` field. You can specify multiple `sort_by` separating them with a comma and specified order ascendant “\_ASC” or descendant “\_DESC” for each field.

- The **limit\_to** parameter allows you to limit the number of results:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]?limit_to=0,10
```

This request returns 10 records beginning at index 0. If you do not specify the index, only the number of results indicated is returned.

The API also returns two custom fields in the header response, “X-LC-QUERY-RESULT” containing the number of results returned in the body response and “X-LC-QUERY-TOTAL” containing the total of records matching your search.

Each record also has a unique URI:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]/[DATA_ID]
```

This request replies a unique record. `[DATA_ID]` must match the unique ID of the record you want to retrieve.

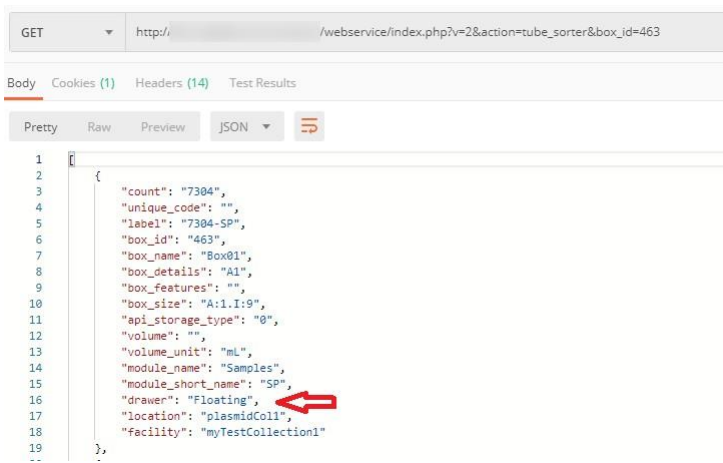
## Storage

You also have **Tube Sorter** filtering functions for every items linked to storage:

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=tube_sorter&box_id=[BOX_ID]
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=tube_sorter&box_id=34`

This request returns storage info on box ID `34` like tube sorter. You can specify multiple IDs by separating them with a comma.



```
GET http://.../webservice/index.php?v=2&action=tube_sorter&box_id=463

Body Cookies (1) Headers (14) Test Results

Pretty Raw Preview JSON

1 {
2   "count": "7304",
3   "unique_code": "",
4   "label": "7304-SP",
5   "box_id": "463",
6   "box_name": "Box01",
7   "box_details": "A1",
8   "box_features": "",
9   "box_size": "A:1.1:9",
10  "api_storage_type": "0",
11  "volume": "",
12  "volume_unit": "ml",
13  "module_name": "Samples",
14  "module_short_name": "SP",
15  "drawer": "Floating",
16  "location": "plasmidColl",
17  "facility": "myTestCollection1"
18 }
19
```

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=tube_sorter&box_id=[BOX_ID]&record_name=[RECORD_NAME]
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=tube_sorter&box_id=206&record_name=ST-260`

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=tube_sorter&record_name=[RECORD_NAME]
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=tube_sorter&record_name=ST-260`

These requests perform filtering on record named ST-260. You can specify multiple record names by separating them with a comma. You can also specify box ID, here 206.

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=tube_sorter&box_name=[BOX_NAME]
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=tube_sorter&box_name=test-rack_06`

This request performs filtering on box `test-rack_06`. You can specify multiple box names by separating them with a comma.

Other search parameters to `action=tube_sorter` can be:

- `location_id`
- `location_name`
- `facility_id`

- `facility_name`

It will return empty boxes as well.

- The `storage_sec` parameter allows retrieving information about secondary storage.

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]&data_id=[DATA_ID]&fields=storage_sec
```

LC local secondary storage information

GET http://localhost/Lab\_5\_422/webservice/index.php?v=2&module=plasmids&data\_id=4&fields=storage\_sec

Params Authorization Headers (10) Body Pre-request Script Tests

Headers (3)

KEY	VALUE
X-LC-APP-Auth	8a4735b2f8ab6b0wc3649cc53c9f
Accept	text/xml
Key	Value

Temporary Headers (7)

Body Cookies (1) Headers (14) Test Results

Pretty Raw Preview XML

```

1 <plasmids>
2 <data>
3 <count>4</count>
4 <storage_sec>
5 <box_id>22</box_id>
6 <unique_code>18</unique_code>
7 <box_details>A2</box_details>
8 <box_name>box2</box_name>
9 <keeper>11</keeper>
10 <volume>200</volume>
11 <volume_unit></volume_unit>
12 </storage_sec>
13 <storage_sec>
14 <box_id>12</box_id>
15 <unique_code>18</unique_code>
16 <box_details>A1</box_details>
17 <box_name>Rack-001.Bx-1</box_name>
18 <keeper>18</keeper>
19 <volume></volume>
20 <volume_unit></volume_unit>
21 </storage_sec>
22 </data>
23 </plasmids>

```

## Product lot

- The `action getLot` allows retrieving lot and reagent info

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=getLot
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=getLot&lot_id=1/LT
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=getLot&chem_id=2
```

Optional parameters are `lot_id` (in format 1 or 1/LT) and `chem_id`. If it doesn't receive parameters, then it retrieves all active lots.

## Recipe

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=getRecipes
```

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=getRecipes
&recipe_id=[record_id]
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=getRecipes
&recipe_id=509`

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=getRecipe
Logs
```

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=getRecipe
Report&log_id=[record_id]
```

e.g. `[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action=getRecipe
Report&log_id=1218`

IDs are examples but are mandatory in these calls.

`getRecipes` prints the following info: id, name, description, category

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** https://.../webservice/index.php?v=2&action=getRecipes
- Query Params:**

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> v	2	
<input checked="" type="checkbox"/> action	getRecipes	
Key	Value	Description
- Body:**

```

1  {
2    {
3      "id": "509",
4      "name": "Second Recipe Test",
5      "description": "Tooltip also in description because 'this description is too big to fit'",
6      "category": "Category I"
7    },
8    {
9      "id": "550",
10     "name": "Recette Pierre",
11     "description": "test pierre",
12     "category": "Category I"

```
- Status:** 200 OK

getRecipes prints the following info for that recipe\_id: id, name, description, category and then the components

GET `https://.../webservice/index.php?v=2&action=getRecipe&recipe_id=509`

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

Query Params

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> v	2	
<input checked="" type="checkbox"/> action	getRecipe	
<input checked="" type="checkbox"/> recipe_id	509	

Body Cookies (1) Headers (12) Test Results 🔗 Status: 200 OK Time

Pretty Raw Preview Visualize JSON ⌵ 🔗

```

1  {
2    "name": "Second Recipe Test",
3    "description": "Tooltip also &iacute;n description because 'this description is too big to fit",
4    "category": "Category I",
5    "components": [
6      {
7        "name": "Extraction Buffer B (EBB)",
8        "quantity": "25.8",
9        "units": "g"
10     },
11     {
12       "name": "Mix de Cellules &#36827;&#21475;&#21407;&#26009;",
13       "quantity": "8",

```

getRecipeLogs prints the following info: id, name, description, category

GET `https://.../webservice/index.php?v=2&action=getRecipeLogs`

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

Query Params

KEY	VALUE
<input checked="" type="checkbox"/> v	2
<input checked="" type="checkbox"/> action	getRecipeLogs
Key	Value

Body Cookies (1) Headers (12) Test Results

Pretty Raw Preview Visualize JSON ⌵ 🔗

```

1  [
2    {
3      "log_id": "1218",
4      "recipe_id": "665",
5      "recipe_name": "gh-Recipe",
6      "usage_date": "2021-11-16 09:08:41",
7      "user": "Wanda Wong"
8    },
9    {
10     "log_id": "1217",
11     "recipe_id": "665",
12     "recipe_name": "gh-Recipe",
13     "usage_date": "2021-11-05 18:13:07",

```

getRecipeReport prints the report PDF for that log\_id under the format base64 that can be decoded into PDF.

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> v	2	
<input checked="" type="checkbox"/> action	getRecipeReport	
<input checked="" type="checkbox"/> log_id	1218	
Key	Value	Description

Body: Cookies (1) Headers (11) Test Results

```

1 JVBEBIXLjCk3eLj9MKOSAwIG91aGo8PCaVvH1zSavUGFn2SavUGFyZw5EIDgMCBSIC9MYXN8TW9kaKZwQzWqE0G6fJAYMTEkMTkXnzI1MTkiMDEnMDAnkSAVuhv2b3vY2VzIDIGM
CBSIC9NZWRpYUJveCBuMC4wMDA0MDgNCA4MDA0MDgMTY4y430DAwMDAgMTESMC41NTEwMDgIC9Dce9wQm94IFswLjAwMDAwMCAwLjAwMDAwMCAwXjgzLjC4MDAwMCAwMTkwLjU1MTAwMFBg1.3Rykw1Cd3g9ZwAuMDA0AIDAwMDA0AIDe200MuNzgaMDA0IDE
x0TAtUxMDA0SAVQXjDQm94IFswLjAwMDAwMCAwLjAwMDAwMCAwXjgzLjC4MDAwMCAwMTkwLjU1MTAwMFBg1.0NvbnR1bnR1ZDEwIDAuIUVu99YXR1IDAgL0dyD3VnIDw0IC9U
eXB1IC9Hce91cCAyUyAvVHJhbnNwYXJzImlmNSIC9DUyAvRGVz2aWn1Uk0cID4+IC9Bbm5vdHMgYwA4IDAgu1BGIc9Qw1AxID4
+CWuZG91aGoMCAwIG91aGo8PCaVvH1zSavUGFn2SavUGFyZw5EIDgMCBSIC9MYXN8TW9kaKZwQzWqE0G6fJAYMTEkMTkXnzI1MTkiMDEnMDAnkSAVuhv2b3vY2VzIDIGM
YHw0Ij0Bz2X07vJdECEIigBf0BZEKTEfCTT1HES5z0dyDIERUAvjAwj040j3lIXj9M6dW+eF0F0vJ7BL7LImQEDHj47A3j2cyY1T61W9u
+BE2TFkndRVoukK207V0UUpEQKqXSeH41e/JHM41f3B2eesR11CnDKIAKjYp7we8SA/UtKhE91QZ1TK1dP6ASvFRuVf64Ez/D-3YcdE65ah1lUsaI3/4VDK644
+ms41qxkt3glzBzR13u7UjHz0Ito8g1VpM511BhAGGmKMQ0w9B1f4JncsjBw1eU2H+Hkjt1Nxn0QdddhG11ABo0QWk0PvEDwUBTQR/ROBwG0UkEULyBy/PtYam/
CzH/QTsghDBo1Fa10B5+PoNum80qH3Q1UkSIRI1qp71lUdQYhN29cMZhLlFbwI4XXH80YG56GHM3QNNhCAMFVKnMTeLkK/9CjNo1ZVEeEN9kNAHu9EFAQBTSCCK
+H2HP1e0VR1N1EXwR50jMS80XxyMnosQm8CjyK231QefZZHk0JTDHF4CT9kGInM141jC3In93Yhks5D0B3adTEHM1UmoJ1xx/
BUZewSuXzb44j1hDc1zcbh06U7Y0Bo18e4FAXclcZwoeS1omI1sJ5Dvm9TYjQKBIYF5vUz1w18Yrj8C5ImHus5P5dXz1C0Xc1WYFvJ4SHVQpBuD18tU01W9y0j3MBLEPv/
a2cx1vT1e1Lkxw906R15k3Vh0LcKAIzH3G27NPG02L1801U0Tae82DhA9eTtznv11aMH01GGK810a128v77nCB959eTKRkuC3E1vao2K3e3uTD1hwZzcxvHGFzow

```

## 2-3-2. POST method

To create a new resource, simply send a request with the POST method to the desired module URI:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]
```

Your parameter keys have to match the field's name.

Check for uniqueness fields have been added, when creating new records (POST) or update (PUT)

If exists a different record with the same value for a field Uniqueness, API will not complete the action and will return code 409 (Conflict), and the text: Value for field 'XXX' must be unique. Value 'YYY' already exists in table 'ZZZ'. (see screenshot)

http://localhost/lab0/webservice/v2/strains


POST http://localhost/lab0/webservice/v2/strains Send

Params Authorization Headers (2) **Body** Pre-request Script Tests Cookies Co

none  form-data  x-www-form-urlencoded  raw  binary

	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	name	new_strain	
<input checked="" type="checkbox"/>	unico	abc	
	Key	Value	Description

Body Cookies (1) Headers (12) Test Results Status: 409 Conflict Time: 365 ms Size: 552 B

Pretty Raw Preview XML 

```
1 Value for field unico must be unique. Value 'abc' already exists in table 'strains'
```

Note: project\_code field can be use in POST and PUT and it expects text (not id). You can now create a new project code if it does not exists and if the operator has permissions enough (administrator or super-administrator).

- The **action addBox** allows you to create a box

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&action= addBox
```

- Required parameters:

- o name
- o type (must be a valid type: box, box\_nogrid, plate, microplate, visotube, bag, shelfpart)
- o equipment (supports id or name and must exist in LabCollector storage).
- o size (depends on the type of box: should be numeric for a visotube, and the format (A:1.H:8) for a box, a plate, and a microplate)

- Optional parameters:

- o description
- o rack
- o keeper

### 2-3-3. PUT method

To modify a resource, simply send a request with the PUT method to the desired record URI:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]/[DATA_ID]
```

Your parameter keys have to match the field's name you want to modify.

For the following actions, note that for PUT request, parameters must be on the body (not in URL).

The URL is `[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2`

The headers are: X-LC-APP-Auth, Accept.

- **Remove Volume**

- Parameters:

- `removeVolume` (mandatory)
- `barcode`, `unique_code` or `aliquot_barcode` (one of them must be present)
- `quantity` (mandatory)

- Response: OK

- **Remove Storage**

- Parameters:

- `removeStorage` (mandatory)
- `barcode`, `unique_code` or `aliquot_barcode` (one of them must be present)

- Response: OK

- **Add Registry Book**

- URL:

```
[PATH_TO_LABCOLLECTOR]/webservice/index.php?v=2&module=[module]
```

- Parameters:

- `addRegistryBook` (mandatory)
- `record_id` (mandatory)
- `date` (mandatory, format yyyy/mm/dd or yyyy-mm-dd)
- `comments` (mandatory)
- `operator` (optional, if it does not send the API default operator will be used)
- `action` (optional, must be a valid 'Storage Action Type' defined in **LC >Admin >Preferences > Process & Actions Type**)

- Response: OK



- **Add Secondary storage**

- Parameters:

- addSecondaryStorage (mandatory)
- barcode (mandatory)
- box\_id (mandatory)
- box\_details (mandatory only for the box with grid divider, tube tray, and microplate. If the box is without grid, a bag, a visotube or a shelf part, it is not required)
- unique\_code (optional)
- volume (optional)
- comments (optional)
- cap\_color (optional)

**Note:** An error message is returned if mandatory parameters are not present; if the barcode does not exist; if the unique\_code is present, but it isn't unique; and, if the color is present but it doesn't exist.

If the parameter box\_details is not received and the type of box needs position (box with grid, tube tray, or microplate), an error message is returned.

KEY	VALUE
<input checked="" type="checkbox"/> addSecondaryStorage	1
<input checked="" type="checkbox"/> unique_code	test_camil_by_api_3
<input checked="" type="checkbox"/> barcode	84/ST
<input checked="" type="checkbox"/> volume	100
<input checked="" type="checkbox"/> box_id	481
<input checked="" type="checkbox"/> box_details	C7,C8
<input checked="" type="checkbox"/> comments	3rd test comment inserting secondary storage via api
<input checked="" type="checkbox"/> cap_color	a00e0e
Key	Value

PUT ▼ http://localhost/labcollector-5.5/webservice/index.php?v=2

Params ● Authorization Headers (11) **Body** ● Pre-request Script Tests Settings

none  form-data  x-www-form-urlencoded  raw  binary  GraphQL BETA

KEY	VALUE
<input checked="" type="checkbox"/> addSecondaryStorage	1
<input checked="" type="checkbox"/> barcode	1/ST
<input checked="" type="checkbox"/> box_id	3
Key	Value

Body ● Cookies (1) Headers (11) Test Results

Pretty Raw Preview Visualize BETA XML ▼ ≡

1 Position in required for this type of box. Please, provide a value for box\_details

Note: project\_code field can be use in POST and PUT and it expects text (not id). You can now create a new project code if it does not exists and if the operator has permissions enough (administrator or super-administrator).

## 2-3-4. DELETE method

To delete a resource, simply send a request with the DELETE method to the desired record URI:

```
[PATH_TO_LABCOLLECTOR]/webservice/v2/[MODULE]/[DATA_ID]
```

### 3- API ERROR MESSAGES

Message	Response code	Description
'Requires application authentication to access the Web Service'	401 Unauthorized	The request either does not have the header parameter X-LC-APP-Auth or does not have a valid value
'Invalid Action xxx'	400 Bad Request	Parameter action has a different value to 'tube_sorter' or 'getBackup'
Missing Search Parameters!	400 Bad Request	The request contains the parameter Action=tube_sorter but it is missing at least one of the following parameters: box_id, box_name, record_name, unique_code, barcode, aliquot_barcode
'Module "XXX" does not exist!'	400 Bad Request	The value of the parameter 'module' is not a LabCollector module
'Module "XXX" does not share this data!'	403 Forbidden	The value of the parameter 'module' is not checked on LabCollector > Admin > Setup > Web service
'The format of the request is not accepted!'	415 Unsupported Media Type	The parameter Accept is used, but the value is not one of the accepted values: application/XML or application/JSON
(Empty)	406 Not Acceptable	The method must be one of the following: GET, POST, PUT, DELETE
'No data found.'	404 Not Found	No data found with this request's parameters
'OK.'	200 OK	Record updated successfully

<b>'Conflict.'</b>	409 Conflict	The record could not be updated because there is a conflict in data
<b>'No organisms value for this module'</b>	404 Not Found	Only the modules "strains", "samples" and "microarrays" have an organism value – you have chosen an incorrect module
<b>'No categories value for this module'</b>	404 Not Found	Only the module 'docs' has categories – you have chosen an incorrect module
<b>'Webservice requires user authentication'</b>	401 Unauthorized	Deprecated
<b>'Your IP is not allowed to access this Web Service'</b>	401 Unauthorized	The client IP is not in the list of authorized IPs for this webservice (LC > Admin > Setup > Web service)
<b>'Error during your request, the following information is mandatory to create a new record: X, Y, Z '</b>	400 Bad Request	Attempt to post new data without mandatory fields X, Y, Z
<b>'An error has occurred during your request, the following information is mandatory to remove volume: unique_code or barcode or aliquot_barcode, quantity, quantity'</b>	400 Bad Request	Attempt to remove volume without mandatory parameters: unique_code or barcode or aliquot_barcode, quantity
<b>'An error has occurred during your request, the following information is mandatory to remove storage: unique_code or barcode or aliquot_barcode, quantity '</b>	400 Bad Request	Attempt to remove storage without a mandatory parameter: unique_code or barcode or aliquot_barcode
<b>"</b>	200 OK	Data requested was returned successfully

## 4- LABCOLLECTOR WEB SERVICE API - ANNEX

The URI system of the API uses a simple and clean URL. Be sure to enable the rewrite engine from Apache to use the URI referenced in the following table.

If the LabCollector server does not support the rewrite engine please use the full URL pattern for your request (secondary URL of each line).

URI		Module	Description
webservice/v2/strains webservice/index.php?v=2&module=strains	<b>GET</b> <b>POST</b>	Strains & Cells	List of all records
webservice/v2/strains/[DATA_ID] webservice/index.php?v=2&module=strains&data_id=[DATA_ID]	<b>GET</b> <b>PUT</b>	Strains & Cells	Unique record
webservice/v2/strains/customfields webservice/index.php?v=2&module=strains&getModuleCustomFields=1	<b>GET</b>	Strains & Cells	List of custom fields
webservice/v2/strains/organisms webservice/index.php?v=2&module=strains&getModuleOrganisms=1	<b>GET</b>	Strains & Cells	List of organisms
webservice/v2/plasmids webservice/index.php?v=2&module=plasmids	<b>GET</b> <b>POST</b>	Plasmids	List of all records
webservice/v2/plasmids/[DATA_ID] webservice/index.php?v=2&module=plasmids&data_id=[DATA_ID]	<b>GET</b> <b>PUT</b>	Plasmids	Unique record
webservice/v2/plasmids/customfields webservice/index.php?v=2&module=plasmids&getModuleCustomFields=1	<b>GET</b>	Plasmids	List of custom fields
webservice/v2/primers webservice/index.php?v=2&module=primers	<b>GET</b> <b>POST</b>	Primers	List of all records
webservice/v2/primers/[DATA_ID] webservice/index.php?v=2&module=primers&data_id=[DATA_ID]	<b>GET</b> <b>PUT</b>	Primers	Unique record
webservice/v2/primers/customfields	<b>GET</b>	Primers	List of custom fields

LABCOLLECTOR WEB SERVICE API - ANNEX

webservice/index.php?v=2&module=primers&getModuleCustomFields=1			
webservice/v2/chemicals webservice/index.php?v=2&module=chemicals	<b>GET POST</b>	Reagents & Supplies	List of all records
webservice/v2/chemicals/[DATA_ID] webservice/index.php?v=2&module=chemicals&data_id=[DATA_ID]	<b>GET PUT</b>	Reagents & Supplies	Unique record
webservice/v2/chemicals/customfields webservice/index.php?v=2&module=chemicals&getModuleCustomFields=1	<b>GET</b>	Reagents & Supplies	List of custom fields
webservice/v2/samples webservice/index.php?v=2&module=samples	<b>GET POST</b>	Samples	List of all records
webservice/v2/samples/[DATA_ID] webservice/index.php?v=2&module=samples&data_id=[DATA_ID]	<b>GET PUT</b>	Samples	Unique record
webservice/v2/samples/customfields webservice/index.php?v=2&module=samples&getModuleCustomFields=1	<b>GET</b>	Samples	List of custom fields
webservice/v2/samples/organisms webservice/index.php?v=2&module=samples&getModuleOrganisms=1	<b>GET</b>	Samples	List of organisms
webservice/v2/samples/types webservice/index.php?v=2&module=samples&getModuleTypes=1	<b>GET</b>	Samples	List of sample types
webservice/v2/antibodies webservice/index.php?v=2&module=antibodies	<b>GET POST</b>	Antibodies	List of all records
webservice/v2/antibodies/[DATA_ID] webservice/index.php?v=2&module=antibodies&data_id=[DATA_ID]	<b>GET PUT</b>	Antibodies	Unique record
webservice/v2/antibodies/customfields webservice/index.php?v=2&module=antibodies&getModuleCustomFields=1	<b>GET</b>	Antibodies	List of custom fields
webservice/v2/sequences webservice/index.php?v=2&module=sequences	<b>GET POST</b>	Sequences	List of all records

LABCOLLECTOR WEB SERVICE API - ANNEX

webservice/v2/sequences/[DATA_ID] webservice/index.php?v=2&module=sequences&data_id=[DATA_ID]	<b>GET PUT</b>	Sequences	Unique record
webservice/v2/sequences/customfields webservice/index.php?v=2&module=sequences&getModuleCustomFields=1	<b>GET</b>	Sequences	List of custom fields
webservice/v2/animals webservice/index.php?v=2&module=animals	<b>GET POST</b>	Animals	List of all records
webservice/v2/animals/[DATA_ID] webservice/index.php?v=2&module=animals&data_id=[DATA_ID]	<b>GET PUT</b>	Animals	Unique record
webservice/v2/animals/customfields webservice/index.php?v=2&module=animals&getModuleCustomFields=1	<b>GET</b>	Animals	List of custom fields
webservice/v2/equipments webservice/index.php?v=2&module=equipments	<b>GET POST</b>	Equipment	List of all records
webservice/v2/equipments/[DATA_ID] webservice/index.php?v=2&module=equipments&data_id=[DATA_ID]	<b>GET PUT</b>	Equipment	Unique record
webservice/v2/equipments/customfields webservice/index.php?v=2&module=equipments&getModuleCustomFields=1	<b>GET</b>	Equipment	List of custom fields
webservice/v2/structures webservice/index.php?v=2&module=structures	<b>GET POST</b>	Chemical Structures	List of all records
webservice/v2/structures/[DATA_ID] webservice/index.php?v=2&module=structures&data_id=[DATA_ID]	<b>GET PUT</b>	Chemical Structures	Unique record
webservice/v2/structures/customfields webservice/index.php?v=2&module=structures&getModuleCustomFields=1	<b>GET</b>	Chemical Structures	List of custom fields
webservice/v2/docs webservice/index.php?v=2&module=docs	<b>GET POST</b>	Documents	List of all records
webservice/v2/docs/[DATA_ID] webservice/index.php?v=2&module=docs&data_id=[DATA_ID]	<b>GET PUT</b>	Documents	Unique record

LABCOLLECTOR WEB SERVICE API - ANNEX

webservice/v2/docs/customfields webservice/index.php?v=2&module=docs &getModuleCustomFields=1	<b>GET</b>	Documents	List of custom fields
webservice/v2/docs/categories webservice/index.php?v=2&module=docs &getModuleCategories=1	<b>GET</b>	Documents	List of categories
webservice/v2/abook webservice/index.php?v=2&module=abook	<b>GET POST</b>	Address Book	List of all records
webservice/v2/abook/[DATA_ID] webservice/index.php?v=2&module=abook &data_id=[DATA_ID]	<b>GET PUT</b>	Address Book	Unique record
webservice/v2/abook/customfields webservice/index.php?v=2&module=abook &getModuleCustomFields=1	<b>GET</b>	Address Book	List of custom fields
webservice/v2/abook/categories webservice/index.php?v=2&module=abook &getModuleCategories=1	<b>GET</b>	Address Book	List of categories
webservice/v2/microarrays webservice/index.php?v=2&module=microarrays	<b>GET POST</b>	Microarrays	List of all records
webservice/v2/microarrays/[DATA_ID] webservice/index.php?v=2&module=microarrays &data_id=[DATA_ID]	<b>GET PUT</b>	Microarrays	Unique record
webservice/v2/microarrays/customfields webservice/index.php?v=2&module=microarrays &getModuleCustomFields=1	<b>GET</b>	Microarrays	List of custom fields
webservice/v2/microarrays/organisms webservice/index.php?v=2&module=microarrays &getModuleOrganisms=1	<b>GET</b>	Microarrays	List of organisms
webservice/v2/[CUSTOM_MODULE_NAME] webservice/index.php?v=2&module=[CUSTOM_MODULE_NAME]	<b>GET POST</b>	Custom Module	List of all records
webservice/v2/[CUSTOM_MODULE_NAME]/[DATA_ID] webservice/index.php?v=2&module=[CUSTOM_MODULE_NAME]&data_id=[DATA_ID]	<b>GET PUT</b>	Custom Module	Unique record



---

webservice/v2/[CUSTOM_MODULE_NAME]/customfields webservice/index.php?v=2&module=[CUSTOM_MODULE_NAME]&getModuleCustomFields=1	<b>GET</b>	Custom Module	List of custom fields
---	------------	---------------	-----------------------



<http://www.labcollector.com>  
[sales@agilebio.com](mailto:sales@agilebio.com)

**AgileBio USA**  
5473 Kearny Villa Road Suite 255  
San Diego, CA 92123  
USA  
Tel: 347 368 1315  
Fax: (800) 453 9128

<http://www.agilebio.com>

**AgileBio Headquarters**  
75 rue de Lourmel  
75015 Paris  
FRANCE  
Tel: 01 41 79 15 85  
Fax: 01 72 70 40 22