

REDCap API Documentation

In order to use the REDCap API for a given REDCap project, you must first be given a token that is specific to your username for that particular project. Rather than using username/passwords, the REDCap API uses tokens as a means of secure authentication, in which a token must be included in every API request. Please note that each user will have a different token for each REDCap project to which they have access. Thus, multiple tokens will be required for making API requests to multiple projects.

To obtain an API token for a project, navigate to that project, then click the API link in the Applications sidebar. On that page you will be able to request an API token for the project from your REDCap administrator, and that page will also display your API token if one has already been assigned. If you do not see a link for the API page on your project's left-hand menu, then someone must first give you API privileges within the project (via the project's User Rights page).

Super Tokens: If you wish to utilize the 'Create Project' API method, then a Super API Token is required. Super tokens are 64-characters in length (as opposed to 32-character-long regular tokens), in which a REDCap user can only have one super token at most. If you are interested in obtaining a super token, please contact your local REDCap administrator.

Introduction to the REDCap API

Provided here is all documentation for utilizing the REDCap API. For general information about the REDCap API or the methods supported by the REDCap API, see the list on the left-hand menu. Each API method provides a specific action that can be performed by REDCap.

The acronym 'API' stands for 'Application Programming Interface'. An API is just a defined way for a program to accomplish a task. The REDCap API is an interface that allows external applications to connect to REDCap remotely, and is used for programmatically retrieving or modifying data or settings within REDCap, such as performing automated data imports/exports for a specified REDCap project. Programmers can use the REDCap API to make applications, websites, apps, widgets, and other projects that interact with REDCap. Programs talk to the REDCap API over HTTP, the same protocol that your browser uses to visit and interact with web pages.

To get started using the REDCap API, begin with the General API Info section on the left-hand menu. There is also an API Playground page in each of your REDCap projects, in which the playground will allow you to experiment with the REDCap API and see the range of its functionality without even having to write any code.

[Obtaining API Tokens](#)

In order to use the REDCap API for a given REDCap project, you must first be given a token that is specific to your username for that particular project. Rather than using username/passwords, the REDCap API uses tokens as a means of secure authentication, in which a token must be included in every API request. Please note that each user will have a different token for each REDCap project to which they have access. Thus, multiple tokens will be required for making API requests to multiple projects.

To obtain an API token for a project, navigate to that project, then click the API link in the Applications sidebar. On that page you will be able to request an API token for the project from your REDCap administrator, and that page will also display your API token if one has already been assigned. If you do not see a link for the API page on your project's left-hand menu, then someone must first give you API privileges within the project (via the project's User Rights page).

Super Tokens: If you wish to utilize the 'Create Project' API method, then a Super API Token is required. Super tokens are 64-characters in length (as opposed to 32-character-long regular tokens), in which a REDCap user can only have one super token at most. If you are interested in obtaining a super token, please contact your local REDCap administrator.

API Error Codes & Responses

HTTP Status Codes:

The REDCap API attempts to return appropriate HTTP status codes for every request. Below is a comprehensive list of all possible HTTP codes that could be returned from the API and what each code signifies.

- **200 OK:** Success!
- **400 Bad Request:** The request was invalid. An accompanying message will explain why.
- **401 Unauthorized:** API token was missing or incorrect.
- **403 Forbidden:** You do not have permissions to use the API.
- **404 Not Found:** The URI you requested is invalid or the resource does not exist.
- **406 Not Acceptable:** The data being imported was formatted incorrectly.
- **500 Internal Server Error:** The server encountered an error processing your request.
- **501 Not Implemented:** The requested method is not implemented.

Error Messages:

When the API returns error messages, it does so in the return format that you specified in your API request (and if not specified, it returns in default format). You can specify the format you want using the **returnFormat** parameter. For example, an error from an XML method might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>

<hash>

  <error>detailed error message</error>

</hash>
```

API Security & Best Practices

Although API requests to REDCap are done using SSL (HTTPS), which means that the traffic to and from the REDCap server is encrypted, there is still more that can be done to ensure the highest level of security when using the API. This is especially important if you are moving sensitive data into or out of REDCap. One thing that is **highly** recommended is for your API script/program (i.e. the thing making the request to the REDCap API) to **validate the SSL certificate of the REDCap web server** when it makes the API request.

Background on SSL certificates:

Web servers have SSL certificates so that their identity can be validated and thus trusted, after which secure, encrypted communication can take place with the server. The reason it is important to validate the server's SSL certificate is because it is possible (although extremely rare) to be the victim of a [Man in the Middle Attack](#) even when your web traffic is secure over SSL/HTTPS. A Man in the Middle (MiM) attack can be performed by a hacker who impersonates the REDCap web server using a fake/invalid SSL certificate. In this way, it is possible for your API script to think that the hacker is really the REDCap server and thus unwittingly send your request not to REDCap but to the hacker, in which he/she can actually see the contents of your request, including your API token, and then use your token to impersonate you to make API requests to REDCap in the future as if they were you.

How to prevent Man in the Middle attacks:

Preventing MiM attacks is pretty simple. Essentially all you need to do is to force your API script to validate the SSL certificate of the REDCap server. REDCap's SSL certificate will always be valid, but the hacker's fake certificate can never be determined to be valid if you attempt to validate it. In many programs or programming languages that can make API requests, validating an SSL certificate is often as easy as setting a flag. For example, [cURL](#) is popularly used by many API scripts in programming languages such as PHP, R, SAS, and many more in order to make the web request to REDCap. **So if your API script is utilizing cURL, all you need to do is modify your script so that it sets the cURL option named CURLOPT_SSL_VERIFYPEER to have a value of TRUE.** Once done, your API script will attempt to make the API request to REDCap **only** if it can validate REDCap's SSL certificate. **Thus by adding the SSL certificate check, you have completely prevented the possibility of MiM attacks and are using the most secure form of communication with the REDCap API.** If you are not using cURL, there are plenty of other examples on the web for how to validate an SSL certificate in different programming languages. Such examples can be found simply by Googling the name of your programming language + 'verify ssl certificate' (e.g., '[Java verify ssl certificate](#)'), which should provide you with many helpful results.

REMINDER:

Please remember that while REDCap itself has many security layers to help protect you and to ensure the highest level of security and data integrity, it is **your** responsibility to ensure that you are using the most secure methods and best practices when using the REDCap API.

[API Examples](#)

The REDCap API can be called from a variety of clients using any popular client-side or web development language that you are able to implement (e.g .NET, Python, PHP, Java). Below you may download a ZIP file containing several examples of how to call the API using various software languages. The files contained therein may be modified however you wish.

[Export Arms](#)

Method Name

Export Arms

Description

This method allows you to export the Arms for a project

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

arm

format

csv, json, xml [default]

Optional

arms

an array of arm numbers that you wish to pull events for (by default, all events are pulled)

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no

format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Arms for the project in the format specified

[Import Arms](#)

Method Name

Import Arms

Description

This method allows you to import Arms into a project or to rename existing Arms in a project. You may use the parameter `override=1` as a 'delete all + import' action in order to erase all existing Arms in the project while importing new Arms. Notice: Because of the 'override' parameter's destructive nature, this method may only use `override=1` for projects in Development status.

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Project Design/Setup privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

arm

override

0 - false [default], 1 - true — You may use `override=1` as a 'delete all + import' action in order to

erase all existing Arms in the project while importing new Arms. If override=0, then you can only add new Arms or rename existing ones.

action

import

format

csv, json, xml [default]

data

Contains the attributes 'arm_num' (referring to the arm number) and 'name' (referring to the arm's name) of each arm to be created/modified, in which they are provided in the specified format.

JSON Example:

```
[{"arm_num": "1", "name": "Drug A"},  
{"arm_num": "2", "name": "Drug B"},  
{"arm_num": "3", "name": "Drug C"}]
```

CSV Example:

arm_num,name

1,Drug A

2,Drug B

3,Drug C

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<arms>
```

```
  <item>
```

```
    <arm_num>1</arm_num>
```

```
    <name>Drug A</name>
```

```
  </item>
```

```
  <item>
```

```
    <arm_num>2</arm_num>
```

```
    <name>Drug B</name>
```

```
  </item>
```

```
  <item>
```

```
    <arm_num>3</arm_num>
```

```
    <name>Drug C</name>
```

</item>

</arms>

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of Arms imported

[Delete Arms](#)

Delete Arms

Description

This method allows you to delete Arms from a project. Notice: Because of this method's destructive nature, it is only available for use for projects in Development status. Additionally, please be aware that deleting an arm also automatically deletes all events that belong to that arm, and will also automatically delete any records/data that have been collected under that arm (this is non-reversible data loss).

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Project Design/Setup privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

arm

action

delete

arms

an array of arm numbers that you wish to delete

Returns

Number of Arms deleted

Data Access Groups**[Export DAGs](#)****Method Name**

Export DAGs

Description

This method allows you to export the Data Access Groups for a project

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges *and* Data Access Groups privileges in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

dag

format

csv, json, xml [default]

Optional**returnFormat**

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

DAGs for the project in the format specified

[Import DAGs](#)

Method Name

Import DAGs

Description

This method allows you to import new DAGs (Data Access Groups) into a project or update the group name of any existing DAGs.

NOTE: DAGs can be renamed by simply changing the group name (data_access_group_name). DAG can be created by providing group name value while unique group name should be set to blank.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Data Access Groups privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

dag

action

import

format

csv, json, xml [default]

data

Contains the attributes 'data_access_group_name' (referring to the group name) and 'unique_group_name' (referring to the auto-generated unique group name) of each DAG to be created/modified, in which they are provided in the specified format.

JSON Example:

```
[{"data_access_group_name":"CA Site","unique_group_name":"ca_site"}  
{"data_access_group_name":"FL Site","unique_group_name":"fl_site"},  
{"data_access_group_name":"New Site","unique_group_name":""}]
```

CSV Example:

data_access_group_name,unique_group_name

"CA Site",ca_site

"FL Site",fl_site

"New Site",

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>  
<dags>  
  <item>  
    <data_access_group_name>CA Site</data_access_group_name>  
    <unique_group_name>ca_site</unique_group_name>  
  </item>  
  <item>  
    <data_access_group_name>FL Site</data_access_group_name>  
    <unique_group_name>fl_site</unique_group_name>  
  </item>  
  <item>  
    <data_access_group_name>New Site</data_access_group_name>  
    <unique_group_name></unique_group_name>  
  </item>  
</dags>
```

Optional**returnFormat**

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no

format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of DAGs added or updated

[Delete DAGs](#)

Method Name

Delete DAGs

Description

This method allows you to delete DAGs from a project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Data Access Groups privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

dag

action

delete

dags

an array of unique group names that you wish to delete

Returns

Number of DAGs deleted

[Switch DAG](#)

Method Name

Switch DAG

Description

This method allows the current API user to switch (assign/reassign/unassign) their current Data Access Group assignment if they have been assigned to multiple DAGs via the DAG Switcher page in the project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

dag

action

switch

dag

The unique group name of the Data Access Group to which you wish to switch.

Returns

Returns "1" when the current API user is switched to the specified Data Access Group, otherwise it returns an error message.

[Export User-DAG Assignment](#)

Method Name

Export User-DAG Assignments

Description

This method allows you to export existing User-DAG assignments for a project

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges *and* Data Access Groups privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

userDagMapping

format

csv, json, xml [default]

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

User-DAG assignments for the project in the format specified

[Import User-DAG Assignment](#)

Method Name

Import User-DAG Assignments

Description

This method allows you to assign users to any data access group.

NOTE: If you wish to modify an existing mapping, you *must* provide its unique username and group name. If the 'redcap_data_access_group' column is not provided, user will not assigned to any group. There should be only one record per username.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Data Access Groups privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

userDagMapping

action

import

format

csv, json, xml [default]

data

Contains the attributes 'username' (referring to the existing unique username) and 'redcap_data_access_group' (referring to existing unique group name) of each User-DAG assignments to be modified, in which they are provided in the specified format.

JSON Example:

```
[{"username":"ca_dt_person","redcap_data_access_group":"ca_site"},  
{"username":"fl_dt_person","redcap_data_access_group":"fl_site"},  
{"username":"global_user","redcap_data_access_group":""}]
```

CSV Example:

username,redcap_data_access_group

ca_dt_person,ca_site

fl_dt_person,fl_site

global_user,

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<items>
```

```
<item>
```

```

    <username>ca_dt_person</username>

    <redcap_data_access_group>ca_site</redcap_data_access_group>
  </item>
  <item>

    <username>fl_dt_person</username>

    <redcap_data_access_group>fl_site</redcap_data_access_group>
  </item>
  <item>

    <username>global_user</username>

    <redcap_data_access_group></redcap_data_access_group>
  </item>
</items>

```

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of User-DAG assignments added or updated

Events

[Export Events](#)

Method Name

Export Events

Description

This method allows you to export the events for a project

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

event

format

csv, json, xml [default]

Optional

arms

an array of arm numbers that you wish to pull events for (by default, all events are pulled)

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Events for the project in the format specified

[Import Events](#)

Method Name

Import Events

Description

This method allows you to import Events into a project or to update existing Events' attributes, such as the event name, days offset, etc. The unique event name of an Event cannot be changed because it is auto-generated by REDCap. Please note that the only way to update an existing Event is to provide the unique_event_name attribute, and if the unique_event_name attribute is missing for an Event being imported (when override=0), it will assume it to be a new Event that should be created. Notice: Because of the 'override' parameter's destructive

nature, this method may only use override=1 for projects in Development status.

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges **and** Project Design/Setup privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

event

action

import

override

0 - false [default], 1 - true — You may use override=1 as a 'delete all + import' action in order to erase all existing Events in the project while importing new Events. If override=0, then you can only add new Events or modify existing ones.

format

csv, json, xml [default]

data

Contains the required attributes 'event_name' (referring to the name/label of the event) and 'arm_num' (referring to the arm number to which the event belongs - assumes '1' if project only contains one arm). In order to modify an existing event, you must provide the attribute 'unique_event_name' (referring to the auto-generated unique event name of the given event). If the project utilizes the Scheduling module, then you may optionally provide the following attributes, which must be numerical: day_offset, offset_min, offset_max. If the day_offset is not provided, then the events will be auto-numbered in the order in which they are provided in the API request.

JSON Example:

```
[{"event_name":"Baseline","arm_num":"1","day_offset":"1","offset_min":"0",  
"offset_max":"0","unique_event_name":"baseline_arm_1"},
```

```
{"event_name":"Visit 1","arm_num":"1","day_offset":"2","offset_min":"0",  
"offset_max":"0","unique_event_name":"visit_1_arm_1"},  
{"event_name":"Visit 2","arm_num":"1","day_offset":"3","offset_min":"0",  
"offset_max":"0","unique_event_name":"visit_2_arm_1"}}
```

CSV Example:

event_name,arm_num

"Baseline",1

"Visit 1",1

"Visit 2",1

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>  
<events>  
  <item>  
    <event_name>Baseline</event_name>  
    <arm_num>1</arm_num>  
  </item>  
  <item>  
    <event_name>Visit 1</event_name>  
    <arm_num>1</arm_num>  
  </item>  
  <item>  
    <event_name>Visit 2</event_name>  
    <arm_num>1</arm_num>  
  </item>  
</events>
```

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of Events imported

[Delete a File](#)

Method Name

Import Events

Description

This method allows you to import Events into a project or to update existing Events' attributes, such as the event name, days offset, etc. The unique event name of an Event cannot be changed because it is auto-generated by REDCap. Please note that the only way to update an existing Event is to provide the `unique_event_name` attribute, and if the `unique_event_name` attribute is missing for an Event being imported (when `override=0`), it will assume it to be a new Event that should be created. Notice: Because of the 'override' parameter's destructive nature, this method may only use `override=1` for projects in Development status.

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Project Design/Setup privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

event

action

import

override

0 - false [default], 1 - true — You may use `override=1` as a 'delete all + import' action in order to erase all existing Events in the project while importing new Events. If `override=0`, then you can only add new Events or modify existing ones.

format

csv, json, xml [default]

data

Contains the required attributes 'event_name' (referring to the name/label of the event) and 'arm_num' (referring to the arm number to which the event belongs - assumes '1' if project only contains one arm). In order to modify an existing event, you must provide the attribute 'unique_event_name' (referring to the auto-generated unique event name of the given event). If the project utilizes the Scheduling module, then you may optionally provide the following attributes, which must be numerical: day_offset, offset_min, offset_max. If the day_offset is not provided, then the events will be auto-numbered in the order in which they are provided in the API request.

JSON Example:

```
[{"event_name":"Baseline","arm_num":"1","day_offset":"1","offset_min":"0",  
"offset_max":"0","unique_event_name":"baseline_arm_1"},  
{"event_name":"Visit 1","arm_num":"1","day_offset":"2","offset_min":"0",  
"offset_max":"0","unique_event_name":"visit_1_arm_1"},  
{"event_name":"Visit 2","arm_num":"1","day_offset":"3","offset_min":"0",  
"offset_max":"0","unique_event_name":"visit_2_arm_1"}]
```

CSV Example:

event_name,arm_num

"Baseline",1

"Visit 1",1

"Visit 2",1

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>  
<events>  
  <item>  
    <event_name>Baseline</event_name>  
    <arm_num>1</arm_num>  
  </item>  
  <item>  
    <event_name>Visit 1</event_name>  
    <arm_num>1</arm_num>  
  </item>
```

```
<item>

  <event_name>Visit 2</event_name>

  <arm_num>1</arm_num>

</item>

</events>
```

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of Events imported

[Delete Events](#)

Method Name

Delete Events

Description

This method allows you to delete Events from a project. Notice: Because of this method's destructive nature, it is only available for use for projects in Development status. Additionally, please be aware that deleting an event will automatically delete any records/data that have been collected under that event (this is non-reversible data loss).

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Project Design/Setup privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each

user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

event

action

delete

events

an array of unique event names that you wish to delete

Returns

Number of Events deleted

Field Names

[Export List of Export Field Names](#)

Method Name

Export List of Export Field Names (i.e. variables used during exports and imports)

Description

This method returns a list of the export/import-specific version of field names for all fields (or for one field, if desired) in a project. This is mostly used for checkbox fields because during data exports and data imports, checkbox fields have a different variable name used than the exact one defined for them in the Online Designer and Data Dictionary, in which *each checkbox option* gets represented as its own export field name in the following format: field_name + triple underscore + converted coded value for the choice. For non-checkbox fields, the export field name will be exactly the same as the original field name. Note: The following field types will be automatically removed from the list returned by this method since they cannot be utilized during the data import process: 'calc', 'file', and 'descriptive'.

The list that is returned will contain the three following attributes for each field/choice: 'original_field_name', 'choice_value', and 'export_field_name'. The choice_value attribute represents the raw coded value for a checkbox choice. For non-checkbox fields, the choice_value attribute will always be blank/empty. The export_field_name attribute represents the export/import-specific version of that field name.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

exportFieldNames

format

csv, json, xml [default]

Optional

field

A field's variable name. By default, all fields are returned, but if field is provided, then it will only the export field name(s) for that field. If the field name provided is invalid, it will return an error.

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'. The list that is returned will contain the original field name (variable) of the field and also the export field name(s) of that field.

Returns

Returns a list of the export/import-specific version of field names for all fields (or for one field, if desired) in a project in the format specified and ordered by their field order . The list that is returned will contain the three following attributes for each field/choice: 'original_field_name', 'choice_value', and 'export_field_name'. The choice_value attribute represents the raw coded value for a checkbox choice. For non-checkbox fields, the choice_value attribute will always be blank/empty. The export_field_name attribute represents the export/import-specific version of that field name.

Files

[Export a File](#)

Method Name

Export a File

Description

This method allows you to download a document that has been attached to an individual record for a File Upload field. Please note that this method may also be used for Signature fields (i.e. File Upload fields with 'signature' validation type).

Note about export rights: Please be aware that Data Export user rights will be applied to this API request. For example, if you have 'No Access' data export rights in the project, then the API file export will fail and return an error. And if you have 'De-Identified' or 'Remove All Identifier Fields' data export rights, then the API file export will fail and return an error *only if* the File

Upload field has been tagged as an Identifier field. To make sure that your API request does not return an error, you should have 'Full Data Set' export rights in the project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

file

action

export

record

the record ID

field

the name of the field that contains the file

event

the unique event name - only for longitudinal projects

repeat_instance

(only for projects with repeating instruments/events) The repeat instance number of the repeating event (if longitudinal) or the repeating instrument (if classic or longitudinal). Default value is '1'.

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

the contents of the file

How to obtain the filename of the file:

The MIME type of the file, along with the name of the file and its extension, can be found in the header of the returned response. Thus in order to determine these attributes of the file being exported, you will need to parse the response header. Example: content-type = application/vnd.openxmlformats-officedocument.wordprocessingml.document; name='FILE_NAME.docx'

[Import a File](#)

Method Name

Import Events

Description

This method allows you to import Events into a project or to update existing Events' attributes, such as the event name, days offset, etc. The unique event name of an Event cannot be changed because it is auto-generated by REDCap. Please note that the only way to update an existing Event is to provide the unique_event_name attribute, and if the unique_event_name attribute is missing for an Event being imported (when override=0), it will assume it to be a new Event that should be created. Notice: Because of the 'override' parameter's destructive nature, this method may only use override=1 for projects in Development status.

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Project Design/Setup privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

event

action
import

override

0 - false [default], 1 - true — You may use override=1 as a 'delete all + import' action in order to erase all existing Events in the project while importing new Events. If override=0, then you can only add new Events or modify existing ones.

format

csv, json, xml [default]

data

Contains the required attributes 'event_name' (referring to the name/label of the event) and 'arm_num' (referring to the arm number to which the event belongs - assumes '1' if project only contains one arm). In order to modify an existing event, you must provide the attribute 'unique_event_name' (referring to the auto-generated unique event name of the given event). If the project utilizes the Scheduling module, then you may optionally provide the following attributes, which must be numerical: day_offset, offset_min, offset_max. If the day_offset is not provided, then the events will be auto-numbered in the order in which they are provided in the API request.

JSON Example:

```
[{"event_name":"Baseline","arm_num":"1","day_offset":"1","offset_min":"0",
"offset_max":"0","unique_event_name":"baseline_arm_1"},
{"event_name":"Visit 1","arm_num":"1","day_offset":"2","offset_min":"0",
"offset_max":"0","unique_event_name":"visit_1_arm_1"},
{"event_name":"Visit 2","arm_num":"1","day_offset":"3","offset_min":"0",
"offset_max":"0","unique_event_name":"visit_2_arm_1"}]
```

CSV Example:

event_name,arm_num

"Baseline",1

"Visit 1",1

"Visit 2",1

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<events>
  <item>
    <event_name>Baseline</event_name>
    <arm_num>1</arm_num>
  </item>
```

```
<item>

  <event_name>Visit 1</event_name>

  <arm_num>1</arm_num>

</item>

<item>

  <event_name>Visit 2</event_name>

  <arm_num>1</arm_num>

</item>

</events>
```

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of Events imported

[Delete a File](#)

Method Name

Delete a File

Description

This method allows you to remove a document that has been attached to an individual record for a File Upload field. Please note that this method may also be used for Signature fields (i.e. File Upload fields with 'signature' validation type).

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

file

action

delete

record

the record ID

field

the name of the field that contains the file

event

the unique event name - only for longitudinal projects

repeat_instance

(only for projects with repeating instruments/events) The repeat instance number of the repeating event (if longitudinal) or the repeating instrument (if classic or longitudinal). Default value is '1'.

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

File Repository

[Create a New Folder in the File Repository](#)

Method Name

Create a New Folder in the File Repository

Description

This method allows you to create a new folder in the File Repository. You may optionally provide the folder_id of the parent folder under which you wish this folder to be created.

Providing a dag_id and/or role_id will allow you to restrict access to only users within a specific DAG (Data Access Group) or User Role, respectively.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges and File Repository privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

fileRepository

action

createFolder

name

The desired name of the folder to be created (max length = 150 characters)

format

csv, json, xml [default]

Optional

folder_id

the folder_id of a specific folder in the File Repository for which you wish to create this sub-folder. If none is provided, the folder will be created in the top-level directory of the File Repository.

dag_id

the dag_id of the DAG (Data Access Group) to which you wish to restrict access for this folder. If none is provided, the folder will be accessible to users in all DAGs and users in no DAGs.

role_id

the role_id of the User Role to which you wish to restrict access for this folder. If none is provided, the folder will be accessible to users in all User Roles and users in no User Roles.

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

The `folder_id` of the new folder created in the specified format. For example, if using `format=json`, the output would look similar to this: `[{"folder_id":45}]`.

[Export a List of Files/Folders from the File Repository](#)

Method Name

Export a List of Files/Folders from the File Repository

Description

This method allows you to export a list of all files and subfolders from a specific folder in a project's File Repository. Each sub-folder will have an associated `folder_id` number, and each file will have an associated `doc_id` number.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges and File Repository privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

fileRepository

action

list

format

csv, json, xml [default]

Optional

folder_id

the folder_id of a specific folder in the File Repository for which you wish to export a list of its files and subfolders. If none is provided, the top-level directory of the File Repository will be used.

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

The list of all files and folders within a given sub-folder in the File Repository in the format specified.

[Export a File from the File Repository](#)**Method Name**

Export a File from the File Repository

Description

This method allows you to download a single file stored in the File Repository by providing the file's doc_id number.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges and File Repository privileges in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

fileRepository

action

export

doc_id

the doc_id of the file in the File Repository

Optional**returnFormat**

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

the contents of the file

[Import a File into the File Repository](#)**Method Name**

Import a File into the File Repository

Description

This method allows you to import a single file into a project's File Repository. The file may be stored in a specific folder in the File Repository if a folder_id is provided.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges and File Repository privileges in the project.

Parameters (case sensitive)**Required**

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

fileRepository

action

import

file

the contents of the file

Optional

folder_id

the folder_id of a specific folder in the File Repository where you wish to store the file. If none is provided, the file will be stored in the top-level directory of the File Repository.

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

[Delete a File from the File Repository](#)

Method Name

Delete a File from the File Repository

Description

This method allows you to delete a single file in a project's File Repository. Once deleted, the file will remain in the Recycle Bin folder for up to 30 days.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges and File Repository privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

fileRepository

action

delete

doc_id

the doc_id of the file in the File Repository

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

[Export Instruments \(Data Entry Forms\)](#)

Method Name

Export Instruments (Data Entry Forms)

Description

This method allows you to export a list of the data collection instruments for a project. This includes their unique instrument name as seen in the second column of the Data Dictionary, as well as each instrument's corresponding instrument label, which is seen on a project's left-hand menu when entering data. The instruments will be ordered according to their order in the project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

instrument

format

csv, json, xml [default]

Returns

Instruments for the project in the format specified and will be ordered according to their order in the project.

[Export PDF file of Instruments](#)

Method Name

Export PDF file of Data Collection Instruments (either as blank or with data)

Description

This method allows you to export a PDF file for any of the following: 1) a single data collection instrument (blank), 2) all instruments (blank), 3) a single instrument (with data from a single record), 4) all instruments (with data from a single record), or 5) all instruments (with data from ALL records). This is the exact same PDF file that is downloadable from a project's data entry form in the web interface, and additionally, the user's privileges with regard to data exports will be applied here just like they are when downloading the PDF in the web interface (e.g., if they have de-identified data export rights, then it will remove data from certain fields in the PDF). If the user has 'No Access' data export rights, they will not be able to use this method, and an error will be returned.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)**Required**

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

pdf

Optional**record**

the record ID. The value is blank by default. If record is blank, it will return the PDF as blank (i.e. with no data). If record is provided, it will return a single instrument or all instruments containing data from that record only.

event

the unique event name - only for longitudinal projects. For a longitudinal project, if record is not blank and event is blank, it will return data for all events from that record. If record is not blank and event is not blank, it will return data only for the specified event from that record.

instrument

the unique instrument name as seen in the second column of the Data Dictionary. The value is blank by default, which returns all instruments. If record is not blank and instrument is blank, it will return all instruments for that record.

repeat_instance

(only for projects with repeating instruments/events) The repeat instance number of the repeating event (if longitudinal) or the repeating instrument (if classic or longitudinal). Default value is '1'.

allRecords

[The value of this parameter does not matter and is ignored.] If this parameter is passed with any value, it will export all instruments (and all events, if longitudinal) with data from all records. Note: If this parameter is passed, the parameters record, event, and instrument will be ignored.

compactDisplay

Set to TRUE to return a compact-formatted PDF that excludes fields that have no data saved and excludes unselected multiple choice options, thus producing a smaller PDF file. If set to FALSE, all fields will be displayed normally.

returnFormat

csv, json, xml [default] - The returnFormat is only used with regard to the format of any error messages that might be returned.

Returns

A PDF file containing one or all data collection instruments from the project, in which the instruments will be blank (no data), contain data from a single record, or contain data from all records in the project, depending on the parameters passed in the API request.

Export Instrument-Event Mappings

Method Name

Export Instrument-Event Mappings

Description

This method allows you to export the instrument-event mappings for a project (i.e., how the data collection instruments are designated for certain events in a longitudinal project).

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

formEventMapping

format

csv, json, xml [default]

Optional

arms

an array of arm numbers that you wish to pull events for (by default, all events are pulled)

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Instrument-event mappings for the project in the format specified

[Import Instrument-Event Mappings](#)

Method Name

Import Instrument-Event Mappings

Description

This method allows you to import Instrument-Event Mappings into a project (this corresponds to the 'Designate Instruments for My Events' page in the project).

NOTE: This only works for longitudinal projects.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Project Design/Setup privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

formEventMapping

format

csv, json, xml [default]

data

Contains the attributes 'arm_num' (referring to the arm number), 'unique_event_name' (referring to the auto-generated unique event name of the given event), and 'form' (referring to the unique form name of the given data collection instrument), in which they are provided in the specified format.

JSON Example:

```
[{"arm_num": "1", "unique_event_name": "baseline_arm_1", "form": "demographics"},  
{"arm_num": "1", "unique_event_name": "visit_1_arm_1", "form": "day_3"}]
```

```
{"arm_num":"1","unique_event_name":"visit_1_arm_1","form":"other"},  
{"arm_num":"1","unique_event_name":"visit_2_arm_1","form":"other"}]
```

CSV Example:

```
arm_num,unique_event_name,form  
1,baseline_arm_1,demographics  
1,visit_1_arm_1,day_3  
1,visit_1_arm_1,other  
1,visit_2_arm_1,other
```

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>  
  
<items>  
  <item>  
    <arm_num>1</arm_num>  
    <unique_event_name>baseline_arm_1</unique_event_name>  
    <form>demographics</form>  
  </item>  
  <item>  
    <arm_num>1</arm_num>  
    <unique_event_name>visit_1_arm_1</unique_event_name>  
    <form>day_3</form>  
  </item>  
  <item>  
    <arm_num>1</arm_num>  
    <unique_event_name>visit_1_arm_1</unique_event_name>  
    <form>other</form>  
  </item>  
  <item>  
    <arm_num>1</arm_num>  
    <unique_event_name>visit_2_arm_1</unique_event_name>  
    <form>other</form>  
  </item>
```

</items>

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of Instrument-Event Mappings imported

Logging

[Export Logging](#)

Method Name

Export Logging

Description

This method allows you to export the logging (audit trail) of all changes made to this project, including data exports, data changes, project metadata changes, modification of user rights, etc.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges *and* Logging privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

log

format

csv, json, xml [default]

Optional**returnFormat**

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

logtype

You may choose event type to fetch result for specific event type

user

To return only the events belong to specific user (referring to existing username), provide a user. If not specified, it will assume all users.

record

To return only the events belong to specific record (referring to existing record name), provide a record. If not specified, it will assume all records. This parameter is available only when event is related to record.

dag

To return only the events belong to specific DAG (referring to `group_id`), provide a dag. If not specified, it will assume all dags.

beginTime

To return only the events that have been logged **after** a given date/time, provide a timestamp in the format YYYY-MM-DD HH:MM (e.g., '2017-01-01 17:00' for January 1, 2017 at 5:00 PM server time). If not specified, it will assume no begin time.

endTime

To return only records that have been logged **before** a given date/time, provide a timestamp in the format YYYY-MM-DD HH:MM (e.g., '2017-01-01 17:00' for January 1, 2017 at 5:00 PM server time). If not specified, it will use the current server time.

Returns

List of all changes made to this project, including data exports, data changes, and the creation or deletion of users.

KEY:

Filter by event (logtype):

export = Data export,

manage = Manage/Design,

user = User or role created-updated-deleted,

record = Record created-updated-deleted,

record_add = Record created (only),

record_edit = Record updated (only),

record_delete = Record deleted (only),
lock_record = Record locking & e-signatures,
page_view = Page Views

Metadata

[Export Metadata \(Data Dictionary\)](#)

Method Name

Export Metadata (Data Dictionary)

Description

This method allows you to export the metadata for a project

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

metadata

format

csv, json, xml [default]

Optional

fields

an array of field names specifying specific fields you wish to pull (by default, all metadata is pulled)

forms

an array of form names specifying specific data collection instruments for which you wish to pull metadata (by default, all metadata is pulled). NOTE: These 'forms' are not the form label values that are seen on the webpages, but instead they are the unique form names seen in Column B of the data dictionary.

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Metadata from the project (i.e. Data Dictionary values) in the format specified ordered by the field order

[Import Metadata \(Data Dictionary\)](#)**Method Name**

Import Metadata (Data Dictionary)

Description

This method allows you to import metadata (i.e., Data Dictionary) into a project. Notice: Because of this method's destructive nature, it is only available for use for projects in Development status.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Project Design/Setup privileges in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

metadata

format

csv, json, xml [default]

data

The formatted data to be imported.

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of fields imported

Projects[Create Project](#)**Method Name**

Create A New Project

Description

This method allows you to create a new REDCap project. A 64-character Super API Token is required for this method (as opposed to project-level API methods that require a regular 32-character token associated with the project-user). In the API request, you must minimally provide the project attributes 'project_title' and 'purpose' (with numerical value 0=Practice/Just for fun, 1=Other, 2=Research, 3=Quality Improvement, 4=Operational Support) when creating a project.

When a project is created with this method, the project will automatically be given all the project-level defaults just as if you created a new empty project via the web user interface, such as a automatically creating a single data collection instrument seeded with a single Record ID field and Form Status field, as well as (for longitudinal projects) one arm with one event. And if you intend to create your own arms or events immediately after creating the project, it is recommended that you utilize the override=1 parameter in the 'Import Arms' or 'Import Events' method, respectively, so that the default arm and event are removed when you add your own. Also, the user creating the project will automatically be added to the project as a user with full user privileges and a project-level API token, which could then be used for subsequent project-level API requests.

NOTE: Only users with Super API Tokens can utilize this method. Users can only be granted a super token by a REDCap administrator (using the API Tokens page in the REDCap Control Center). Please be advised that users with a Super API Token can create new REDCap projects via the API without any approval needed by a REDCap administrator. If you are interested in obtaining a super token, please contact your local REDCap administrator.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have a Super API Token.

Parameters (case sensitive)

Required

token

The Super API Token specific to a user

content

project

format

csv, json, xml [default]

data

Contains the attributes of the project to be created, in which they are provided in the specified format. While the only required attributes are 'project_title' and 'purpose', the fields listed below are all the possible attributes that can be provided in the 'data' parameter. The 'purpose' attribute must have a numerical value (0=Practice/Just for fun, 1=Other, 2=Research, 3=Quality Improvement, 4=Operational Support), in which 'purpose_other' is only required to have a value (as a text string) if purpose=1. The attributes is_longitudinal (0=False, 1=True; Default=0), surveys_enabled (0=False, 1=True; Default=0), and record_autonumbering_enabled (0=False, 1=True; Default=1) are all boolean. Please note that either is_longitudinal=1 or surveys_enabled=1 does not add arms/events or surveys to the project, respectively, but it merely enables those settings which are seen at the top of the project's Project Setup page.

All available attributes:

project_title, purpose, purpose_other, project_notes, is_longitudinal, surveys_enabled, record_autonumbering_enabled

JSON Example:

```
[{"project_title":"My New REDCap Project","purpose":"0"}]
```

CSV Example:

```
project_title,purpose,is_longitudinal
```

```
"My New REDCap Project",0,1
```

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<item>
```

```
  <project_title>My New REDCap Project</project_title>
```

```
  <purpose>0</purpose>
```

```
  <surveys_enabled>1</surveys_enabled>
```

```
  <record_autonumbering_enabled>0</record_autonumbering_enabled>
```

```
</item>
```

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

odm

default: NULL - The 'odm' parameter must be an XML string in CDISC ODM XML format that contains project metadata (fields, forms, events, arms) and might optionally contain data to be imported as well. The XML contained in this parameter can come from a REDCap Project XML export file from REDCap itself, or may come from another system that is capable of exporting projects and data in CDISC ODM format. If the 'odm' parameter is included in the API request, it will use the XML to import its contents into the newly created project. This will allow you not only to create the project with the API request, but also to import all fields, forms, and project attributes (and events and arms, if longitudinal) as well as record data all at the same time.

Returns

When a project is created, a 32-character project-level API Token is returned (associated with both the project and user creating the project). This token could then ostensibly be used to make subsequent API calls to this project, such as for adding new events, fields, records, etc.

[Import Project Info](#)

Method Name

Import Project Information

Description

This method allows you to update some of the basic attributes of a given REDCap project, such as the project's title, if it is longitudinal, if surveys are enabled, etc. Its data format corresponds to the format in the API method Export Project Information.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Project Setup/Design privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

project_settings

format

csv, json, xml

data

Contains some or all of the attributes from Export Project Information in the same data format as in the export. These attributes will change the project information.

Attributes for the project in the format specified. For any values that are boolean, they should be represented as either a '0' (no/false) or '1' (yes/true). The following project attributes can be updated:

project_title, project_language, purpose, purpose_other, project_notes, custom_record_label, secondary_unique_field, is_longitudinal, surveys_enabled, scheduling_enabled, record_autonumbering_enabled, randomization_enabled, project_irb_number, project_grant_number, project_pi_firstname, project_pi_lastname, display_today_now_button, bypass_branching_erase_field_prompt

Returns

Returns the number of values accepted to be updated in the project settings (including values which remained the same before and after the import).

[Export Project Info](#)**Method Name**

Export Project Information

Description

This method allows you to export some of the basic attributes of a given REDCap project, such as the project's title, if it is longitudinal, if surveys are enabled, the time the project was created and moved to production, etc.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)**Required**

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

project

format

csv, json, xml [default]

Optional**returnFormat**

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Attributes for the project in the format specified. For any values that are boolean, they will be represented as either a '0' (no/false) or '1' (yes/true). Also, all date/time values will be returned in Y-M-D H:M:S format. The following attributes will be returned:

project_id, project_title, creation_time, production_time, in_production, project_language, purpose, purpose_other, project_notes, custom_record_label, secondary_unique_field, is_longitudinal, has_repeating_instruments_or_events, surveys_enabled, scheduling_enabled, record_autonumbering_enabled, randomization_enabled, ddp_enabled, project_irb_number, project_grant_number, project_pi_firstname, project_pi_lastname, display_today_now_button, missing_data_codes, external_modules, bypass_branching_erase_field_prompt

[Export Project XML](#)**Method Name**

Export Entire Project as REDCap XML File (containing metadata & data)

Description

The entire project (all records, events, arms, instruments, fields, and project attributes) can be downloaded as a single XML file, which is in CDISC ODM format (ODM version 1.3.1). This XML file can be used to create a clone of the project (including its data, optionally) on this REDCap server or on another REDCap server (it can be uploaded on the Create New Project page). Because it is in CDISC ODM format, it can also be used to import the project into another ODM-compatible system. NOTE: All the option parameters listed below ONLY apply to data returned if the 'returnMetadataOnly' parameter is set to FALSE (default). For this API method, ALL metadata (all fields, forms, events, and arms) will always be exported. Only the data returned can be filtered using the optional parameters.

Note about export rights: If the 'returnMetadataOnly' parameter is set to FALSE, then please be aware that Data Export user rights will be applied to any data returned from this API request. For example, if you have 'De-Identified' or 'Remove All Identifier Fields' data export rights, then some data fields *might* be removed and filtered out of the data set returned from the API. To make sure that no data is unnecessarily filtered out of your API request, you should have 'Full Data Set' export rights in the project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

project_xml

Optional

returnMetadataOnly

true, false [default] - TRUE returns only metadata (all fields, forms, events, and arms), whereas FALSE returns all metadata and also data (and optionally filters the data according to any of the optional parameters provided in the request)

records

an array of record names specifying specific records you wish to pull (by default, all records are pulled)

fields

an array of field names specifying specific fields you wish to pull (by default, all fields are pulled)

events

an array of unique event names that you wish to pull records for - only for longitudinal projects

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

exportSurveyFields

true, false [default] - specifies whether or not to export the survey identifier field (e.g.,

'redcap_survey_identifier') or survey timestamp fields (e.g., instrument+'_timestamp') when surveys are utilized in the project. If you do not pass in this flag, it will default to 'false'. If set to 'true', it will return the redcap_survey_identifier field and also the survey timestamp field for a particular survey when at least one field from that survey is being exported. NOTE: If the survey identifier field or survey timestamp fields are imported via API data import, they will simply be ignored since they are not real fields in the project but rather are pseudo-fields.

exportDataAccessGroups

true, false [default] - specifies whether or not to export the 'redcap_data_access_group' field when data access groups are utilized in the project. If you do not pass in this flag, it will default to 'false'. NOTE: This flag is only viable if the user whose token is being used to make the API request is **not** in a data access group. If the user is in a group, then this flag will revert to its default value.

filterLogic

String of logic text (e.g., [age] > 30) for filtering the data to be returned by this API method, in which the API will only return the records (or record-events, if a longitudinal project) where the logic evaluates as TRUE. This parameter is blank/null by default unless a value is supplied. Please note that if the filter logic contains any incorrect syntax, the API will respond with an error message.

exportFiles

true, false [default] - TRUE will cause the XML returned to include all files uploaded for File Upload and Signature fields for all records in the project, whereas FALSE will cause all such fields not to be included. NOTE: Setting this option to TRUE can make the export very large and may prevent it from completing if the project contains many files or very large files.

Returns

The entire REDCap project's metadata (and data, if specified) will be returned in CDISC ODM format as a single XML string.

Records

[Export Records](#)

Method Name

Export Records

Description

This method allows you to export a set of records for a project.

Note about export rights: Please be aware that Data Export user rights will be applied to this API request. For example, if you have 'No Access' data export rights in the project, then the API data export will fail and return an error. And if you have 'De-Identified' or 'Remove All Identifier Fields' data export rights, then some data fields **might** be removed and filtered out of the data set returned from the API. To make sure that no data is unnecessarily filtered out of your API request, you should have 'Full Data Set' export rights in the project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

record

format

csv, json, xml [default], odm ('odm' refers to CDISC ODM XML format, specifically ODM version 1.3.1)

type

- flat - output as one record per row [default]
- eav - output as one data point per row
 - Non-longitudinal: Will have the fields - record*, field_name, value
 - Longitudinal: Will have the fields - record*, field_name, value, redcap_event_name

* 'record' refers to the record ID for the project

Optional

records

an array of record names specifying specific records you wish to pull (by default, all records are pulled)

fields

an array of field names specifying specific fields you wish to pull (by default, all fields are pulled) or alternatively as a string (comma-separated list).

forms

an array of form names you wish to pull records for. If the form name has a space in it, replace the space with an underscore (by default, all records are pulled)

events

an array of unique event names that you wish to pull records for - only for longitudinal projects

rawOrLabel

raw [default], label - export the raw coded values or labels for the options of multiple choice fields

rawOrLabelHeaders

raw [default], label - (for 'csv' format 'flat' type only) for the CSV headers, export the variable/field names (raw) or the field labels (label)

exportCheckboxLabel

true, false [default] - specifies the format of checkbox field values specifically when exporting the data as labels (i.e., when rawOrLabel=label) in flat format (i.e., when type=flat). When exporting labels, by default (without providing the exportCheckboxLabel flag or if exportCheckboxLabel=false), all checkboxes will either have a value 'Checked' if they are checked or 'Unchecked' if not checked. But if exportCheckboxLabel is set to true, it will instead export the checkbox value as the checkbox option's label (e.g., 'Choice 1') if checked or it will be blank/empty (no value) if not checked. If rawOrLabel=false or if type=eav, then the exportCheckboxLabel flag is ignored. (The exportCheckboxLabel parameter is ignored for type=eav because 'eav' type always exports checkboxes differently anyway, in which checkboxes are exported with their true variable name (whereas the 'flat' type exports them as variable___code format), and another difference is that 'eav' type **always** exports checkbox values as the choice label for labels export, or as 0 or 1 (if unchecked or checked, respectively) for raw export.)

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

exportSurveyFields

true, false [default] - specifies whether or not to export the survey identifier field (e.g., 'redcap_survey_identifier') or survey timestamp fields (e.g., instrument+'_timestamp') when surveys are utilized in the project. If you do not pass in this flag, it will default to 'false'. If set to 'true', it will return the redcap_survey_identifier field and also the survey timestamp field for a particular survey when at least one field from that survey is being exported. NOTE: If the survey identifier field or survey timestamp fields are imported via API data import, they will simply be ignored since they are not real fields in the project but rather are pseudo-fields.

exportDataAccessGroups

true, false [default] - specifies whether or not to export the 'redcap_data_access_group' field when data access groups are utilized in the project. If you do not pass in this flag, it will default to 'false'. NOTE: This flag is only viable if the user whose token is being used to make the API request is **not** in a data access group. If the user is in a group, then this flag will revert to its default value.

filterLogic

String of logic text (e.g., [age] > 30) for filtering the data to be returned by this API method, in which the API will only return the records (or record-events, if a longitudinal project) where the logic evaluates as TRUE. This parameter is blank/null by default unless a value is supplied. Please note that if the filter logic contains any incorrect syntax, the API will respond with an error message.

dateRangeBegin

To return only records that have been created or modified **after** a given date/time, provide a timestamp in the format YYYY-MM-DD HH:MM:SS (e.g., '2017-01-01 00:00:00' for January 1, 2017 at midnight server time). If not specified, it will assume no begin time.

dateRangeEnd

To return only records that have been created or modified **before** a given date/time, provide a timestamp in the format YYYY-MM-DD HH:MM:SS (e.g., '2017-01-01 00:00:00' for January 1, 2017 at midnight server time). If not specified, it will use the current server time.

csvDelimiter

Set the delimiter used to separate values in the CSV data file (for CSV format only). Options include: comma ',' (default), 'tab', semi-colon ';', pipe '|', or caret '^'. Simply provide the value in quotes for this parameter.

decimalCharacter

If specified, force all numbers into same decimal format. You may choose to force all data values containing a decimal to have the same decimal character, which will be applied to all calc fields and number-validated text fields. Options include comma ',' or dot/full stop '.', but if left blank/null, then it will export numbers using the fields' native decimal format. Simply provide the value of either ',' or '.' for this parameter.

exportBlankForGrayFormStatus

true, false [default] - specifies whether or not to export blank values for instrument complete status fields that have a gray status icon. All instrument complete status fields having a gray icon can be exported either as a blank value or as "0" (Incomplete). Blank values are recommended in a data export if the data will be re-imported into a REDCap project.

Returns

Data from the project in the format and type specified ordered by the record (primary key of project) and then by event id

EAV XML:

```
<?xml version="1.0" encoding="UTF-8" ?>

<records>

  <item>

    <record></record>

    <field_name></field_name>

    <value></value>

    <redcap_event_name></redcap_event_name>

  </item>

</records>
```

Flat XML:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

<records>

<item>

each data point as an element

...

</item>

</records>

[Import Records](#)

Method Name

Import Records

Description

This method allows you to import a set of records for a project

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

record

format

csv, json, xml [default], odm ('odm' refers to CDISC ODM XML format, specifically ODM version 1.3.1)

type

- flat - output as one record per row [default]
- eav - input as one data point per row
 - Non-longitudinal: Will have the fields - record*, field_name, value

- Longitudinal: Will have the fields - record*, field_name, value, redcap_event_name

* 'record' refers to the record ID for the project

** Event name is the unique name for an event, not the event label

overwriteBehavior

- normal - blank/empty values will be ignored [default]
- overwrite - blank/empty values are valid and will overwrite data

forceAutoNumber

If record auto-numbering has been enabled in the project, it may be desirable to import records where each record's record name is automatically determined by REDCap (just as it does in the user interface). If this parameter is set to 'true', the record names provided in the request will not be used (although they are still required in order to associate multiple rows of data to an individual record in the request), but instead those records in the request will receive new record names during the import process. NOTE: To see how the provided record names get translated into new auto record names, the returnContent parameter should be set to 'auto_ids', which will return a record list similar to 'ids' value, but it will have the new record name followed by the provided record name in the request, in which the two are comma-delimited. For example, if

- false (or 'false') - The record names provided in the request will be used. [default]
- true (or 'true') - New record names will be automatically determined.

backgroundProcess

Specifies whether to do the import as background process.

- 0 or 'false' for no. [default]
- 1 or 'true' for yes.

data

The formatted data to be imported.

TIP: If importing repeating instances for a repeating event or repeating instrument, you may auto-number the instances by providing a value of 'new' for the 'redcap_repeat_instance' field in the dataset you are importing. This is useful because it allows you to import such data without the need to determine how many instances already exist for a given repeating event/instance prior to the import. **NOTICE:** The 'new' value option for auto-numbering instances does NOT work for 'eav' type data but only for 'flat' type.

NOTE: When importing data in EAV type format, please be aware that checkbox fields must have their field_name listed as variable+'___'+optionCode and its value as either '0' or '1' (unchecked or checked, respectively). For example, for a checkbox field with variable name 'icecream', it would be imported as EAV with the field_name as 'icecream___4' having a value of '1' in order to set the option coded with '4' (which might be 'Chocolate') as 'checked'.

EAV XML:

```
<?xml version="1.0" encoding="UTF-8" ?>

<records>

  <item>

    <record></record>

    <field_name></field_name>

    <value></value>

    <redcap_event_name></redcap_event_name>

  </item>

</records>
```

Flat XML:

```
<?xml version="1.0" encoding="UTF-8" ?>

<records>

  <item>

    each data point as an element

    ...

  </item>

</records>
```

Optional

dateFormat

MDY, DMY, YMD [default] - the format of values being imported for dates or datetime fields (understood with M representing 'month', D as 'day', and Y as 'year') - NOTE: The default format is Y-M-D (with **dashes**), while MDY and DMY values should always be formatted as M/D/Y or D/M/Y (with **slashes**), respectively.

csvDelimiter

Set the delimiter used to separate values in the CSV data file (for CSV format only). Options include: comma ',' (default), 'tab', semi-colon ';', pipe '|', or caret '^'. Simply provide the value in quotes for this parameter.

returnContent

count [default] - the number of records imported, ids - a list of all record IDs that were imported, auto_ids = (used only when forceAutoNumber=true) a list of pairs of all record IDs that were imported, includes the new ID created and the ID value that was sent in the API request (e.g., 323,10).

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

the content specified by **returnContent**

Delete Records**Method Name**

Delete Records

Description

This method allows you to delete one or more records from a project in a single API request, and also optionally allows you to delete parts of a record, such as a single instrument's data for one or more records or a single event's data for one or more records.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have 'Delete Record' user privileges in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

record

action

delete

records

an array of record names specifying specific records you wish to delete

Optional

arm

the arm number of the arm in which the record(s) should be deleted. (This can only be used if the project is longitudinal with more than one arm.) NOTE: If the arm parameter is not provided, the specified records will be deleted from all arms in which they exist. Whereas, if arm is provided, they will only be deleted from the specified arm.

instrument

the unique instrument name (column B in the Data Dictionary) of an instrument (as a string) if you wish to delete the data for all fields on the specified instrument for the records specified.

event

the unique event name - only for longitudinal projects. NOTE: If instrument is provided for a longitudinal project, the event parameter is mandatory.

repeat_instance

the repeating instance number for a repeating instrument or repeating event. NOTE: If project has repeating instruments/events, it will remove only the data for that repeating instance.

delete_logging

provide a value of '0' ("keep logging") or '1' ("delete logging"). This option can only be utilized for projects with the "Delete a record's logging activity when deleting the record?" setting enabled by an administrator on the Edit Project Settings page. If that setting is enabled in the project, then the default value will be '1', and if the project-level setting is not enabled, the default value will be '0'. If the project-level setting has been enabled, this option must be provided with a value of '0' in order to prevent the record's logging activity from being deleted when the record is deleted.

Returns

the number of records deleted or (if instrument, event, and/or instance are provided) the number of items deleted over the total records specified;

Rename Record**Method Name**

Rename Record

Description

This method allows you to rename a record from a project in a single API request.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have 'Rename Record' user privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

record

action

rename

record

record name of the current record which you want to rename to new name.

new_record_name

new record name to which you want to rename current record.

Optional

arm

specific arm number in which current record exists. If null, then all records with same name across all arms on which it exists (if longitudinal with multiple arms) will be renamed to new record name, otherwise it will rename the record only in the specified arm.

Returns

Returns "1" if record is renamed or error message if any.

[Generate Next Record Name](#)

Method Name

Generate Next Record Name

Description

To be used by projects with record auto-numbering enabled, this method exports the next potential record ID for a project. It generates the next record name by determining the current maximum numerical record ID and then incrementing it by one.

Note: This method does not create a new record, but merely determines what the next record name would be.

If using Data Access Groups (DAGs) in the project, this method accounts for the special formatting of the record name for users in DAGs (e.g., DAG-ID); in this case, it only assigns the next value for ID for all numbers inside a DAG. For example, if a DAG has a corresponding DAG number of 223 wherein records 223-1 and 223-2 already exist, then the next record will be 223-3 if the API user belongs to the DAG that has DAG number 223. (The DAG number is auto-assigned by REDCap for each DAG when the DAG is first created.) When generating a new record name in a DAG, the method considers all records in the entire project when determining the maximum record ID, including those that might have been originally created in that DAG but then later reassigned to another DAG.

Note: This method functions the same even for projects that do not have record auto-numbering enabled.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

generateNextRecordName

Returns

The maximum integer record ID + 1.

[Randomize Record](#)**Method Name**

Randomize Record

Description

This method allows the current API user to randomize a record.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method you must have the Randomize privilege in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each

user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

record

action

randomize

record

The record name (id) of the record to randomize. The record must already exist and contain all necessary stratification information.

randomization_id

The unique id of the randomization (viewable on the Randomization page for users with Design permissions, or on the API Playground page). Corresponds to a specific target field and event.

format

csv, json, xml [default], odm ('odm' refers to CDISC ODM XML format, specifically ODM version 1.3.1)

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

returnAlt

false [default], true - return the value for the alternative target field, i.e. the randomization number for open allocations. Note: with concealed allocations only the value '*' will be returned, not the real allocation group (which would break the blinding).

Returns

Performs the specified randomization for the record and returns the value for the target randomization field (plus optionally the alternative target value), or an error message on failure (such as if the record does not exist or if stratification information is missing).

JSON Example:

```
{
  "randomization_id": 123,
  "record": "21",
  "target_field": "1",
  "target_field_alt": "R-1005"
}
```

CSV Example:

randomization_id,record,target_field,target_field_alt

123,21,1,R-1005

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<item>
  <randomization_id>123</randomization_id>
  <record>21</record>
  <target_field>1</target_field>
  <target_field_alt>R-1005</target_field_alt>
</item>
```

Repeating Instruments and Events

[Export Repeating Instruments and Events](#)

Method Name

Export Repeating Instruments and Events

Description

This method allows you to export a list of the repeated instruments and repeating events for a project. This includes their unique instrument name as seen in the second column of the Data Dictionary, as well as each repeating instrument's corresponding custom repeating instrument label. For longitudinal projects, the unique event name is also returned for each repeating instrument. Additionally, repeating events are returned as separate items, in which the instrument name will be blank/null to indicate that it is a repeating event (rather than a repeating instrument).

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges *and* Project Setup/Design privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

repeatingFormsEvents

format

csv, json, xml [default]

Returns

Repeated instruments and events for the project in the format specified and will be ordered according to their order in the project.

[Import Repeating Instruments and Events](#)**Method Name**

Import Repeating Instruments and Events

Description

This method allows you to import a list of the repeated instruments and repeating events for a project. This includes their unique instrument name as seen in the second column of the Data Dictionary, as well as each repeating instrument's corresponding custom repeating instrument label. For longitudinal projects, the unique event name is also needed for each repeating instrument. Additionally, repeating events must be submitted as separate items, in which the instrument name will be blank/null to indicate that it is a repeating event (rather than a repeating instrument).

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* Project Design/Setup privileges in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

repeatingFormsEvents

format

csv, json, xml [default]

data

Note: Super API Tokens can also be utilized for this method instead of a project-level API token.

Users can only be granted a super token by a REDCap administrator (using the API Tokens page in the REDCap Control Center).

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of repeated instruments or repeated events that have been imported

Reports

[Export Reports](#)

Method Name

Export Reports

Description

This method allows you to export the data set of a report created on a project's 'Data Exports, Reports, and Stats' page.

Note about export rights: Please be aware that Data Export user rights will be applied to this API request. For example, if you have 'No Access' data export rights in the project, then the API report export will fail and return an error. And if you have 'De-Identified' or 'Remove All Identifier Fields' data export rights, then some data fields *might* be removed and filtered out of the data set returned from the API. To make sure that no data is unnecessarily filtered out of your API request, you should have 'Full Data Set' export rights in the project.

Also, please note the the 'Export Reports' method does *not* make use of the 'type' (flat/eav) parameter, which can be used in the 'Export Records' method. All data for the 'Export Reports' method is thus exported in flat format. If the 'type' parameter is supplied in the API request, it will be ignored.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

report

report_id

the report ID number provided next to the report name on the report list page

format

csv, json, xml [default], odm ('odm' refers to CDISC ODM XML format, specifically ODM version 1.3.1)

Optional**returnFormat**

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

rawOrLabel

raw [default], label - export the raw coded values or labels for the options of multiple choice fields

rawOrLabelHeaders

raw [default], label - (for 'csv' format 'flat' type only) for the CSV headers, export the variable/field names (raw) or the field labels (label)

exportCheckboxLabel

true, false [default] - specifies the format of checkbox field values specifically when exporting the data as labels (i.e., when `rawOrLabel=label`). When exporting labels, by default (without providing the `exportCheckboxLabel` flag or if `exportCheckboxLabel=false`), all checkboxes will either have a value 'Checked' if they are checked or 'Unchecked' if not checked. But if `exportCheckboxLabel` is set to true, it will instead export the checkbox value as the checkbox option's label (e.g., 'Choice 1') if checked or it will be blank/empty (no value) if not checked. If `rawOrLabel=false`, then the `exportCheckboxLabel` flag is ignored.

csvDelimiter

Set the delimiter used to separate values in the CSV data file (for CSV format only). Options include: comma ',' (default), 'tab', semi-colon ';', pipe '|', or caret '^'. Simply provide the value in quotes for this parameter.

decimalCharacter

If specified, force all numbers into same decimal format. You may choose to force all data values containing a decimal to have the same decimal character, which will be applied to all calc fields and number-validated text fields. Options include comma ',' or dot/full stop '.', but if

left blank/null, then it will export numbers using the fields' native decimal format. Simply provide the value of either ',' or '.' for this parameter.

Returns

Data from the project in the format and type specified ordered by the record (primary key of project) and then by event id

[Export REDCap Version](#)

Method Name

Export REDCap Version

Description

This method returns the current REDCap version number as plain text (e.g., 4.13.18, 5.12.2, 6.0.0).

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

Note: Super API Tokens can also be utilized for this method instead of a project-level API token. Users can only be granted a super token by a REDCap administrator (using the API Tokens page in the REDCap Control Center).

content

version

format

csv, json, xml [default]

Returns

the current REDCap version number (three numbers delimited with two periods) as plain text - e.g., 4.13.18, 5.12.2, 6.0.0

Surveys

[Export a Survey Link](#)

Method Name

Export a Survey Link for a Participant

Description

This method returns a unique survey link (i.e., a URL) in plain text format for a specified record and data collection instrument (and event, if longitudinal) in a project. If the user does not have 'Survey Distribution Tools' privileges, they will not be able to use this method, and an error will be returned. If the specified data collection instrument has not been enabled as a survey in the project, an error will be returned.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

surveyLink

record

the record ID. The name of the record in the project.

instrument

the unique instrument name as seen in the second column of the Data Dictionary. This instrument must be enabled as a survey in the project.

event

the unique event name (for longitudinal projects only).

repeat_instance

(only for projects with repeating instruments/events) The repeat instance number of the repeating event (if longitudinal) or the repeating instrument (if classic or longitudinal). Default value is '1'.

Optional

returnFormat

csv, json, xml [default] - The returnFormat is only used with regard to the format of any error messages that might be returned.

Returns

Returns a unique survey link (i.e., a URL) in plain text format for the specified record and instrument (and event, if longitudinal).

[Export a Survey Access Code](#)**Method Name**

Export a Survey Access Code for a Participant

Description

This method returns a survey access code for a specified record and data collection instrument (and event, if longitudinal) in a project. If the user does not have 'Survey Distribution Tools' privileges, they will not be able to use this method, and an error will be returned. If the specified data collection instrument has not been enabled as a survey in the project, an error will be returned.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

surveyAccessCode

record

the record ID. The name of the record in the project.

instrument

the unique instrument name as seen in the second column of the Data Dictionary. This instrument must be enabled as a survey in the project.

event

the unique event name (for longitudinal projects only).

repeat_instance

(only for projects with repeating instruments/events) The repeat instance number of the repeating event (if longitudinal) or the repeating instrument (if classic or longitudinal). Default value is '1'.

Optional**returnFormat**

csv, json, xml [default] - The returnFormat is only used with regard to the format of any error messages that might be returned.

Returns

Returns a survey access code for the specified record and instrument (and event, if longitudinal).

[Export Survey Participants](#)**Method Name**

Export a Survey Participant List

Description

This method returns the list of all participants for a specific survey instrument (and for a specific event, if a longitudinal project). If the user does not have 'Survey Distribution Tools' privileges, they will not be able to use this method, and an error will be returned. If the specified data collection instrument has not been enabled as a survey in the project, an error will be returned.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges *and* Survey Distribution Tools privileges in the project.

Parameters (case sensitive)**Required****token**

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

participantList

instrument

the unique instrument name as seen in the second column of the Data Dictionary. This instrument must be enabled as a survey in the project.

event

the unique event name (for longitudinal projects only).

format

csv, json, xml [default]

Optional**returnFormat**

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Returns the list of all participants for the specified survey instrument [and event] in the desired format. The following fields are returned: email, email_occurrence, identifier, invitation_sent_status, invitation_send_time, response_status, survey_access_code, survey_link. The attribute 'email_occurrence' represents the current count that the email address has appeared in the list (because emails can be used more than once), thus email + email_occurrence represent a unique value pair. 'invitation_sent_status' is '0' if an invitation has not yet been sent to the participant, and is '1' if it has. 'invitation_send_time' is the date/time in which the next invitation will be sent, and is blank if there is no invitation that is scheduled to be sent. 'response_status' represents whether the participant has responded to the survey, in which its value is 0, 1, or 2 for 'No response', 'Partial', or 'Completed', respectively. Note: If an incorrect event_id or instrument name is used or if the instrument has not been enabled as a survey, then an error will be returned.

[Export a Survey Queue Link](#)**Method Name**

Export a Survey Queue Link for a Participant

Description

This method returns a unique Survey Queue link (i.e., a URL) in plain text format for the specified record in a project that is utilizing the Survey Queue feature. If the user does not have 'Survey Distribution Tools' privileges, they will not be able to use this method, and an error will be returned. If the Survey Queue feature has not been enabled in the project, an error will be returned.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

surveyQueueLink

record

the record ID. The name of the record in the project.

Optional

returnFormat

csv, json, xml [default] - The returnFormat is only used with regard to the format of any error messages that might be returned.

Returns

Returns a unique Survey Queue link (i.e., a URL) in plain text format for the specified record in the project.

[Export a Survey Return Code](#)

Method Name

Export a Survey Return Code for a Participant

Description

This method returns a unique Return Code in plain text format for a specified record and data collection instrument (and event, if longitudinal) in a project. If the user does not have 'Survey Distribution Tools' privileges, they will not be able to use this method, and an error will be returned. If the specified data collection instrument has not been enabled as a survey in the project or does not have the 'Save & Return Later' feature enabled, an error will be returned.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

surveyReturnCode

record

the record ID. The name of the record in the project.

instrument

the unique instrument name as seen in the second column of the Data Dictionary. This instrument must be enabled as a survey in the project.

event

the unique event name (for longitudinal projects only).

repeat_instance

(only for projects with repeating instruments/events) The repeat instance number of the repeating event (if longitudinal) or the repeating instrument (if classic or longitudinal). Default value is '1'.

Optional

returnFormat

csv, json, xml [default] - The returnFormat is only used with regard to the format of any error messages that might be returned.

Returns

Returns a unique Return Code in plain text format for the specified record and instrument (and event, if longitudinal).

Users & User Privileges

[Export Users](#)

Method Name

Export Users

Description

This method allows you to export the list of users for a project, including their user privileges and also email address, first name, and last name. Note: If the user has been assigned to a user role, it will return the user with the role's defined privileges.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges *and* User Rights privileges (Full Access or Read Only) in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

user

format

csv, json, xml [default]

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

The method will return all the attributes below with regard to user privileges in the format specified. Please note that the 'forms' attribute is the only attribute that contains sub-elements (one for each data collection instrument), in which each form will have its own Form Rights value (see the key below to learn what each numerical value represents). Most user privilege attributes are boolean (0=No Access, 1=Access). Attributes returned:

username, email, firstname, lastname, expiration, data_access_group, design, alerts, user_rights, data_access_groups, data_export, reports, stats_and_charts, manage_survey_participants, calendar, data_import_tool, data_comparison_tool, logging, email_logging, file_repository, data_quality_create, data_quality_execute, api_export, api_import, api_modules, mobile_app, mobile_app_download_data, record_create, record_rename, record_delete, lock_records_customization, lock_records, lock_records_all_forms, forms, forms_export

KEY:

Data Export: 0=No Access, 2=De-Identified, 3=Remove Identifier Fields, 1=Full Data Set

Form Rights: 0=No Access, 2=Read Only, 1=View records/responses and edit records (survey responses are read-only), 3=Edit survey responses

Other attribute values: 0=No Access, 1=Access.

[Import Users](#)

Method Name

Import Users

Description

This method allows you to import new users into a project while setting their user privileges, or update the privileges of existing users in the project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* User Rights privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

user

format

csv, json, xml [default]

data

Contains the attributes of the user to be added to the project or whose privileges in the project are being updated, in which they are provided in the specified format. All values should be numerical with the exception of username, expiration, data_access_group, and forms. Please note that the 'forms' attribute is the only attribute that contains sub-elements (one for each data collection instrument), in which each form will have its own Form Rights value (see the key below to learn what each numerical value represents). Most user privilege attributes are boolean (0=No Access, 1=Access). Please notice the distinction between data_access_group (contains the unique DAG name for a user) and data_access_groups (denotes whether the user has access to the Data Access Groups page).

Missing attributes: If a user is being added or re-imported to a project in the API request, then any attributes not provided for a user in the request (including form-level rights) will automatically be given the minimum privileges (typically 0=No Access) for the attribute/privilege. However, if an existing user's privileges are being modified in the API request, then any attributes not provided will not be modified from their current value but only

the attributes provided in the request will be modified.

Data Export: 0=No Access, 2=De-Identified, 3=Remove Identifier Fields, 1=Full Data Set

Form Rights: 0=No Access, 2=Read Only, 1=View records/responses and edit records (survey responses are read-only), 3=Edit survey responses

Other attribute values: 0=No Access, 1=Access.

All available attributes:

username, expiration, data_access_group, design, alerts, user_rights, data_access_groups, data_export, reports, stats_and_charts, manage_survey_participants, calendar, data_import_tool, data_comparison_tool, logging, email_logging, file_repository, data_quality_create, data_quality_execute, api_export, api_import, api_modules, mobile_app, mobile_app_download_data, record_create, record_rename, record_delete, lock_records_customization, lock_records, lock_records_all_forms, forms, forms_export

JSON Example:

```
{
  "username": "harrispa",
  "expiration": "",
  "data_access_group": "",
  "design": "1",
  "user_rights": "1",
  "data_access_groups": "1",
  "data_export": "1",
  "reports": "1",
  "stats_and_charts": "1",
  "manage_survey_participants": "1",
  "calendar": "1",
  "data_import_tool": "1",
  "data_comparison_tool": "1",
  "logging": "1",
  "file_repository": "1",
  "data_quality_create": "1",
  "data_quality_execute": "1",
  "api_export": "1",
  "api_import": "1",
  "api_modules": "1",
  "mobile_app": "1",
  "mobile_app_download_data": "0",
  "record_create": "1",
  "record_rename": "0",
  "record_delete": "0",
  "lock_records_all_forms": "0",
  "lock_records": "0",
  "lock_records_customization": "0",
  "forms": {
    "demographics": "1",
    "day_3": "1",
    "other": "1"
  },
  "username": "taylorr4",
  "expiration": "2015-12-07",
  "data_access_group": "",
  "design": "0",
  "user_rights": "0",
  "data_access_groups": "0",
  "data_export": "2",
  "reports": "1",
  "stats_and_charts": "1",
  "manage_survey_participants": "1",
  "calendar": "1",
  "data_import_tool": "0",
  "data_comparison_tool": "0",
  "logging": "0",
  "file_repository": "1",
  "data_quality_create": "0",
  "data_quality_execute": "0",
  "api_export": "0",
  "api_import": "0",
  "api_modules": "0",
  "mobile_app": "0",
  "mobile_app_download_data": "0",
  "record_create": "1",
  "record_rename": "0",
  "record_delete": "0",
  "lock_records_all_forms": "0",
  "lock_records": "0",
  "lock_records_customization": "0",
  "forms": {
    "demographics": "1",
    "day_3": "2",
    "other": "0"
  },
  "forms_export": {
    "demographics": "1",
    "day_3": "0",
    "other": "2"
  }
}
```

CSV Example:

username,design,user_rights,forms,forms_export

harrispa,1,1,"demographics:1,day_3:1,other:1","demographics:1,day_3:0,other:2"

taylorr4,0,0,"demographics:1,day_3:2,other:0","demographics:1,day_3:2,other:0"

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<users>
```

```
  <item>
```

```
    <username>harrispa</username>
```

```
    <expiration>2015-12-07</expiration>
```

```
    <user_rights>1</user_rights>
```

```
    <design>0</design>
```

```
    <forms>
```

```
      <demographics>1</demographics>
```

```
      <day_3>2</day_3>
```

```
      <other>0</other>
```

```
    </forms>
```

```
    <forms_export>
```

```
      <demographics>1</demographics>
```

```
      <day_3>0</day_3>
```

```
      <other>2</other>
```

```
    </forms_export>
```

```
  </item>
```

```
</users>
```

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of users added or updated

[Delete Users](#)

Method Name

Delete Users

Description

This method allows you to delete Users from a project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* User Rights privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

user

action

delete

users

an array of unique usernames that you wish to delete

Returns

Number of Users deleted

User Roles

[Export User Roles](#)

Method Name

Export User Roles

Description

This method allows you to export the list of user roles for a project, including their user privileges.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges *and* User Rights privileges (Full Access or Read Only) in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

userRole

format

csv, json, xml [default]

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

The method will return all the attributes below with regard to user roles privileges in the format specified. Please note that the 'forms' attribute is the only attribute that contains sub-elements (one for each data collection instrument), in which each form will have its own Form Rights value (see the key below to learn what each numerical value represents). Most user role privilege attributes are boolean (0=No Access, 1=Access). Attributes returned:

unique_role_name, role_label, design, alerts, user_rights, data_access_groups, reports, stats_and_charts, manage_survey_participants, calendar, data_import_tool, data_comparison_tool, logging, email_logging, file_repository, data_quality_create, data_quality_execute, api_export, api_import, api_modules, mobile_app, mobile_app_download_data, record_create, record_rename, record_delete, lock_records_customization, lock_records, lock_records_all_forms, forms, forms_export

KEY:

Data Export: 0=No Access, 2=De-Identified, 3=Remove Identifier Fields, 1=Full Data Set

Form Rights: 0=No Access, 2=Read Only, 1=View records/responses and edit records (survey

responses are read-only), 3=Edit survey responses
Other attribute values: 0=No Access, 1=Access.

[Import User Roles](#)

Method Name

Import User Roles

Description

This method allows you to import new user roles into a project while setting their privileges, or update the privileges of existing user roles in the project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* User Rights privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

userRole

format

csv, json, xml [default]

data

Contains the attributes of the user role to be added to the project or whose privileges in the project are being updated, in which they are provided in the specified format. All values should be numerical with the exception of unique_role_name, role_label and forms. Please note that the 'forms' attribute is the only attribute that contains sub-elements (one for each data collection instrument), in which each form will have its own Form Rights value (see the key below to learn what each numerical value represents). Most user privilege attributes are boolean (0=No Access, 1=Access).

Missing attributes: If a user role is being added to a project in the API request, then any attributes not provided for a user role in the request (including form-level rights) will automatically be given the minimum privileges (typically 0=No Access) for the attribute/privilege. However, if an existing user role's privileges are being modified in the API request, then any attributes not provided will not be modified from their current value but only

the attributes provided in the request will be modified.

Data Export: 0=No Access, 2=De-Identified, 3=Remove Identifier Fields, 1=Full Data Set
Form Rights: 0=No Access, 2=Read Only, 1=View records/responses and edit records (survey responses are read-only), 3=Edit survey responses
Other attribute values: 0=No Access, 1=Access.

All available attributes:

unique_role_name, role_label, design, alerts, user_rights, data_access_groups, reports, stats_and_charts, manage_survey_participants, calendar, data_import_tool, data_comparison_tool, logging, email_logging, file_repository, data_quality_create, data_quality_execute, api_export, api_import, api_modules, mobile_app, mobile_app_download_data, record_create, record_rename, record_delete, lock_records_customization, lock_records, lock_records_all_forms, forms, forms_export

JSON Example:

```
{
  "unique_role_name": "U-2119C4Y87T",
  "role_label": "Project Manager",
  "design": "0",
  "user_rights": "1",
  "data_access_groups": "1",
  "data_export_tool": "0",
  "reports": "1",
  "stats_and_charts": "1",
  "manage_survey_participants": "1",
  "calendar": "1",
  "data_import_tool": "0",
  "data_comparison_tool": "0",
  "logging": "0",
  "file_repository": "1",
  "data_quality_create": "0",
  "data_quality_execute": "0",
  "api_export": "0",
  "api_import": "0",
  "api_modules": "0",
  "mobile_app": "0",
  "mobile_app_download_data": "0",
  "record_create": "1",
  "record_rename": "0",
  "record_delete": "0",
  "lock_records_customization": "0",
  "lock_records": "0",
  "lock_records_all_forms": "0",
  "forms": {
    "demographics": "1",
    "day_3": "2",
    "other": "0"
  },
  "unique_role_name": "U-527D39JXAC",
  "role_label": "Data Entry Person",
  "design": "0",
  "user_rights": "0",
  "data_access_groups": "0",
  "data_export_tool": "0",
  "reports": "1",
  "stats_and_charts": "1",
  "manage_survey_participants": "1",
  "calendar": "1",
  "data_import_tool": "0",
  "data_comparison_tool": "0",
  "logging": "0",
  "file_repository": "1",
  "data_quality_create": "0",
  "data_quality_execute": "0",
  "api_export": "1",
  "api_import": "0",
  "api_modules": "0",
  "mobile_app": "0",
  "mobile_app_download_data": "0",
  "record_create": "1",
  "record_rename": "0",
  "record_delete": "0",
  "lock_records_customization": "0",
  "lock_records": "0",
  "lock_records_all_forms": "0",
  "forms": {
    "demographics": "1",
    "day_3": "1",
    "other": "0"
  },
  "forms_export": {
    "demographics": "1",
    "day_3": "2",
    "other": "1"
  }
}
```

CSV Example:

unique_role_name,role_label,design,user_rights,forms,forms_export

U-527D39JXAC,Data Entry

Person,1,1,"demographics:1,day_3:1,other:1","demographics:1,day_3:2,other:0"

U-2119C4Y87T,Project

Manager,0,0,"demographics:1,day_3:2,other:0","demographics:1,day_3:2,other:0"

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<users>
```

```
  <item>
```

```
    <unique_role_name>U-527D39JXAC</unique_role_name>
```

```
    <role_label>Data Entry Person</role_label>
```

```
    <user_rights>1</user_rights>
```

```
    <design>0</design>
```

```
    <forms>
```

```
      <demographics>1</demographics>
```

```
      <day_3>2</day_3>
```

```
      <other>0</other>
```

```
    </forms>
```

```
    <forms_export>
```

```
      <demographics>1</demographics>
```

```
      <day_3>0</day_3>
```

```
      <other>2</other>
```

```
    </forms>
```

```
  </item>
```

```
</users>
```

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of user roles added or updated

[Delete User Roles](#)

Method Name

Delete User Roles

Description

This method allows you to delete User Roles from a project.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* User Rights privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

userRole

action

delete

roles

an array of unique rolenames that you wish to delete

Returns

Number of User Roles deleted

[Export User-Role Assignment](#)

Method Name

Export User-Role Assignments

Description

This method allows you to export existing User-Role assignments for a project

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Export privileges *and* User Rights privileges (Full Access or Read Only) in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

userRoleMapping

format

csv, json, xml [default]

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., backgroundProcess=true). When using a background process, success:true (upon success) or success:false (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

User-Role assignments for the project in the format specified

[Import User-Role Assignment](#)

Method Name

Import User-Role Assignments

Description

This method allows you to assign users to any user role.

NOTE: If you wish to modify an existing mapping, you **must** provide its unique username and role name. If the 'unique_role_name' column is not provided, user will not assigned to any user role. There should be only one record per username.

URL

<https://redcap.hc.fm.usp.br/api/>

Supported Request Method

POST

Permissions Required

To use this method, you must have API Import/Update privileges *and* User Rights privileges in the project.

Parameters (case sensitive)

Required

token

The API token specific to your REDCap project and username (each token is unique to each user for each project). See the section on the left-hand menu for obtaining a token for a given project.

content

userRoleMapping

action

import

format

csv, json, xml [default]

data

Contains the attributes 'username' (referring to the existing unique username) and 'unique_role_name' (referring to existing unique role name) of each User-Role assignments to be modified, in which they are provided in the specified format. Note: If the project contains Data Access Groups, you may pass a third attribute 'data_access_group' with a valid unique DAG name of the DAG to which you wish to assign the user during the process of assigning them to a user role.

JSON Example:

```
[{"username":"global_user","unique_role_name":""},
{"username":"ca_dt_person","unique_role_name":"U-2119C4Y87T"},
{"username":"fl_dt_person","unique_role_name":"U-2119C4Y87T"}]
```

CSV Example:

```
username,unique_role_name
ca_dt_person,U-2119C4Y87T
fl_dt_person,U-2119C4Y87T
global_user,
```

XML Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<items>
  <item>
```

```
<username>ca_dt_person</username>

<unique_role_name>U-2119C4Y87T</unique_role_name>

</item>

<item>

  <username>fl_dt_person</username>

  <unique_role_name>U-2119C4Y87T</unique_role_name>

</item>

<item>

  <username>global_user</username>

  <unique_role_name></unique_role_name>

</item>

</items>
```

Optional

returnFormat

csv, json, xml - specifies the format of error messages. If you do not pass in this flag, it will select the default format for you passed based on the 'format' flag you passed in or if no format flag was passed in, it will default to 'xml'.

NOTE: Does not apply when importing as a background process (i.e., `backgroundProcess=true`). When using a background process, `success:true` (upon success) or `success:false` (upon failure) will be returned in the appropriate format (csv, json, xml).

Returns

Number of User-Role assignments added or updated