

ConfiForms Server REST API version 0

- [Using ConfiForms REST API](#)
- [Create entry](#)
- [Update entry](#)
- [Delete entry](#)
- [Delete entries](#)
- [Update specific field](#)
- [Validate entry](#)
- [Search entries](#)
- [Get by ID](#)
- [Get form definition](#)
- [Upload file to ConfiForms field](#)

Using ConfiForms REST API

Here is the list of REST endpoints and supported parameters for ConfiForms app

Please refer to [Confluence page permissions and ConfiForms](#) to understand the security model used by ConfiForms (ConfiForms works in the same security / authentication context as your Confluence - same users / same authentication scheme).

Both methods are supported:

- POST (for file uploads only POST)
- GET



pageld is a very important parameter and should be supplied always, as all the methods are executed when PAGE is known/defined, otherwise a NOT FOUND error is shown

See Atlassian guide how to find pageld parameter for the page - <https://confluence.atlassian.com/confkb/how-to-get-confluence-page-id-648380445.html>



Adding "**_skipIfParamsEmpty**" parameter with value "true" to your request helps to bypass setting all the fields - useful when you have field definitions with validations and you don't want to set / reset those with your REST API call

```
_skipIfParamIsEmpty=true
```



Adding "**skipIFTTTExecute**" parameter with value "true" to your request tells ConfiForms form to skip IFTTT rules execution

```
skipIFTTTExecute=true
```

This parameter is supported by the following methods

- [Create entry](#)
- [Update entry](#)
- [Delete entry](#)
- [Delete entries](#)
- [Update specific field](#)

Make sure you enable exports on the form if you plan to use an account that does not have form administrative permissions ([Confluence page permissions and ConfiForms](#))

Data Export tricks and hints

For form non-admin users (see here who is form administrator and who is not - [Confluence page permissions and ConfiForms](#)) you need to explicitly allow data exports. This is a security measure to prevent forms data from being exported by anyone when you don't want to.

To enable exports for non admin users you need to allow exports in the Form Definition for your form

Insert 'ConfiForms Form Definition' Macro

☒ Allow data export and enable search

If you want to enable data export and access via REST APIs for form's non-admin users (You will still need to set "Enable export" parameter on the TableView macro). This also enables form's data to be searchable via Confluence search

 Preview

And then you can enable the option to export the data in the corresponding views

For example, this is how you do it for the ConfiForms TableView

Insert 'ConfiForms TableView' Macro

values

☒ Enable export

ConfiForms data in CSV, XLS, XML and JSON format

 Preview

Method / URL

Create entry

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/rest/save.action?pageId=<PAGE_ID>&f=<FORM_NAME>&fields...
```

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

fields... - fields here is a set of key-value pairs for ConfiForms form's fields you want to set (separated with '&' as HTTP parameters). For example if the form has a field with a name 'bookTitle' defined then you will need to write bookTitle=Book Title

Since version 2.12 you can POST fields as JSON payload (instead of setting fields in the URL via &fields...). There structure is as follows

```
"fields": {
  "field1": "field value",
  "field2": "2020-04-25"
}
```



There is a way to create entries on behalf of the "page" - useful when you have "autopage" fields in your form

In this case, use the "storagePageId" as the page location of your form, while "pageId" parameter use as the "current page"

Update entry

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/rest/save.action?pageId=<PAGE_ID>&f=<FORM_NAME>&entryId=ID&fields...
```

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

ID - entity id you would like to update, if not given then a new entity is created

fields... - fields here is a set of key-value pairs for ConfiForms form's fields you want to set (separated with '&' as HTTP parameters). For example if the form has a field with a name 'bookTitle' defined then you will need to write bookTitle=Book Title

Delete entry

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/delete.action?
pageId=<PAGE_ID>&registrationFormName=<FORM_NAME>&entryId=<ID>
```

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

ID - entity id you would like to delete

Delete entries

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/remove.action?pageId=<PAGE_ID>&f=<FORM_NAME>&q=<FILTER>
```

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

FILTER - filter to match records you want to delete. Same syntax as the filter itself uses. See [Using filters](#) section in documentation

Update specific field

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/rest/update.action?pageId=<PAGE_ID>&f=<FORM_NAME>&q=<FILTER>&fv=field:
value
```

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

FILTER - filter to match records you want to update. Same syntax as the filter itself uses. See [Using filters](#) section in documentation

fv=field:value - parameter fv shows which form's field to update and it's new value



From ConfiForms version 1.49.2 you can use [entry.fieldname] in "fv" parameter (to reference the value from another field)

```
fv=field:[entry.some_other_field]
```

Validate entry

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/rest/validate.action?pageId=<PAGE_ID>&f=<FORM_NAME>&fields...
```

This method only validates if the data could be saved

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

fields... - fields here is a set of key-value pairs for ConfiForms form's fields you want to set (separated with '&' as HTTP parameters). For example if the form has a field with a name 'bookTitle' defined then you will need to write bookTitle=Book Title

Search entries

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/rest/filter.action?pageId=<PAGE_ID>&f=<FORM_NAME>&q=<QUERY>
```

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

QUERY - search query in Lucene format, For example to search for books which have a title ('bookTitle' field on the form) set to 'Catching fire' you would need to write the following expression: &q=bookTitle:Catching fire.

Take a look at [Using filters](#) tutorial to learn more by example

Optional: limitFields - to specify the fields to return (metadata fields are always returned!)

New in **VERSION 3.5.5** introduction of *ignoreCase* parameter. You can set the filtering to ignore case when matching (default) or not to ignore case when filtering entries, setting ignoreCase=false



From ConfiForms version 1.49.2 you can use additional "fields" parameter in the request to decorate the returned values with [Virtual functions](#). Use "," to separate field decoration instructions

```
fields=myfield.camelCase,mydatefield.formatDate(dd-MM-yyyy)
```

This translates to - try to apply camelCase virtual function on "myfield" and try to format "mydatefield" with the formatDate function and given pattern



From ConfiForms version 1.50, you can specify limits (**limit**) and sorting (**sort**)

```
limit=2&sort=created ASC
```

Sorting parameter expects the value as described here [Sorting in ConfiForms](#)



From ConfiForms version 1.53.3 you can use use "append=true" in your query which instructs the parameter "fields" to add the transformed (decorated) fields into the record.

This allows you to have multiple representations of the field in the output.

Get by ID

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/rest/get.action?pageId=<PAGE_ID>&f=<FORM_NAME>&id=<UUID>
```

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

UUID - record ID

Optional: limitFields - to specify the fields to return (metadata fields are always returned!)

AVAILABLE SINCE 2.12

Get form definition

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/rest/definition.action?pageId=<PAGE_ID>&f=<FORM_NAME>
```

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

Upload file to ConfiForms field

Since ConfiForms version 1.20 you are now able to attach files to ConfiForms records via REST API. Make sure you create a POST request and for larger files please consider using a multipart request instead, having file body sent via the file part, not via base64 encoded contents. In this case, do not set the "parameter!"

File should be encoded as BASE64 string

```
http(s)://CONFLUENCE_SERVER/ajax/confiforms/rest/upload.action?
pageId=<PAGE_ID>&f=<FORM_NAME>&q=<FILTER>&fv=<FIELD_NAME>:
<BASE64ENCODED_STRING>&fileContentType=<CONTENT_TYPE>&fileName=<NAMEOFFILE>
```

PAGE_ID - where the ConfiForms form is configured

FORM_NAME - name of ConfiForms form

FILTER - filter to find the entry you want to upload this file to (for example: id:GUID, id:39554293-f4c6-4ae5-9d34-cb34db1f44a0 to upload to a particular entry, filter can be anything ConfiForms supports [ConfiForms Filters](#))

FIELD_NAME - name of the file followed by : and file contents encoded as BASE64 string

CONTENT_TYPE - file content type which you are trying to upload, expecting mime type, see <https://www.sitepoint.com/web-foundations/mime-types-complete-list/>

NAMEOFFILE - name of the file (will be visible as the name in attachments)



It is strongly advised to use **POST** for this operation and send the parameters not in the query string but using the HTTP POST method request parameters

For larger files consider using a **multi-part POST** request and sending your file as a multi-part part, not as encoded base64 contents!

Assuming your form is called "myform", it is located on page "819211" and you want to upload a file to field "photo" and attach it to the record that matches the filter "name:alex". Here is what you shall do (screenshot is based on POSTMAN UI)

POST		http://localhost:1990/confluence/ajax/confiforms/rest/upload.action	
Params Authorization Headers (11) Body Pre-request Script Tests Settings			
none form-data x-www-form-urlencoded raw binary GraphQL BETA			
	KEY	VALUE	
<input checked="" type="checkbox"/>	photo	me.png X	
<input checked="" type="checkbox"/>	pageId	819211	
<input checked="" type="checkbox"/>	f	myform	
<input checked="" type="checkbox"/>	fv	photo:	
<input checked="" type="checkbox"/>	q	name:alex	
<input checked="" type="checkbox"/>	fileContentType	image/png	
<input checked="" type="checkbox"/>	fileName	alex.png	