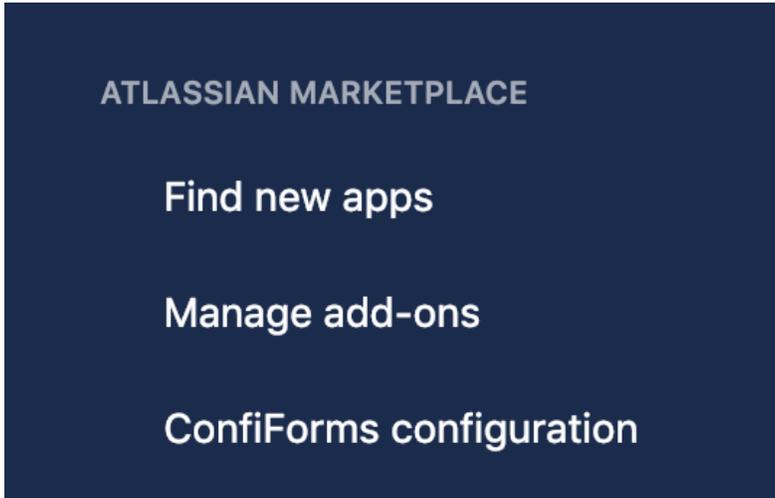


REST API for ConfiForms CLOUD

This is a documentation for ConfiForms CLOUD REST API v1

To start with API you will need to generate an access token.

This is done in ConfiForms settings. Go to your Confluence settings and browse to "Atlassian Marketplace" section



Click on "ConfiForms configuration"

Select "API tokens" tab and click on "Generate token" button to generate an access token for ConfiForms REST API

ConfiForms configuration

[General settings](#) [Webservices connections](#) [API tokens](#)

API tokens for ConfiForms REST API

Here you can configure access tokens for ConfiForms REST APIs

Token	Expiration date	Status	
xxx-xxx-xxx-xxx-xxx-xxx-xxx	2020-01-08 11:50:24.021	VALID	Delete

[Generate token](#)

Generated token is valid for 1 year and shall be used to access APIs

ConfiForms REST APIs

Server URL is <https://app.confiforms.net>

- [Search](#)
- [Delete](#)
- [Validate](#)
- [Save](#)
- [Get](#)
- [Update](#)
- [Definition](#)
- [Examples](#)

API prefix **/rest/api/v1/**

 Please note that {token} can be omitted in the URLs if it is given via HTTP **request header** called "*Auth-Token*"

Search	Searches and filters ConfiForms entries based QUERY given	GET	<p><code>/search/{token}/{pageld}/{formName}?q=<QUERY></code></p> <p>Supports optional parameters:</p> <ul style="list-style-type: none"> sort - Sorting in ConfiForms startAt - starts with 1 for first record if not specified. Must be a positive numeric value and larger than 0 limit - number of records to show per page. Maximum (and the default) number is 200 records per page 	Where QUERY is an expression in ConfiForms Filters format
Delete	Deletes ConfiForms record from the form by id (UUID)	DELETE	<code>/delete/{token}/{pageld}/{formName}/{entryId}</code>	User must have permissions to edit/delete the record. This means ConfiForms cloud system user (com.vertuna.confluence.plugins.confiforms) must have enough permissions to do that.
Validate	Validates parameters against the rules and field types configured by the form	GET	<code>/validate/{token}/{pageld}/{formName}?<FIELDS></code>	fields are given in a key=value format as any HTTP request parameters
Save	Saves the parameters given as new record in ConfiForms form. All the rules are applied before saving according to form configuration	GET POST	<p><code>/save/{token}/{pageld}/{formName}?<FIELDS></code></p> <p>Instead of <FIELDS> parameters added to the request url query string you can use POST method and send over the JSON payload with the structure as follows:</p> <pre>{ "fields": { "field1": "field value", "field2": "2020-04-25" } }</pre> <p>Multi-value fields you can give as JSON arrays</p> <pre>"myfield": ["value1", "value2"]</pre>	<p>fields are given in a key=value format as any HTTP request parameters</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> <p> When you need to update the record the ID shall be given as "entryId" parameter</p> <p>When no entryId parameter found or the record by given entryId could not be found then a new record will be created</p> </div> <p>Partial updates are supported. Meaning that you can supply less parameters than the record has (missing fields will not be updated).</p> <p>To set the field value to empty, do supply it as a parameter</p> <p>myfield=</p> <p>(where "myfield" is the name of the field you want to set the empty value for)</p>
Get	Loads ConfiForms record by id (UUID)	GET	<code>/get/{token}/{pageld}/{formName}/{entryId}</code>	404 if not found
Update	Updates ConfiForms field by query (for each record matching the query)	GET POST	<code>/update/{token}/{pageld}/{formName}?q=<QUERY>&fv=field:value</code>	Where QUERY is an expression in ConfiForms Filters format and "fv" is given in a field:value format, where "field" is the name of the field in ConfiForms record you want to update with "value"
Definition	Returns form configuration	GET	<code>/definition/{token}/{pageld}/{formName}</code>	Returns form definition (configuration) 404 if not found

Examples

Example result returned by "Search"

`/rest/api/v1/search/<TOKEN>/553943100/f?q=ve*`

```

{
  "entries": [
    {
      "recordId": 0,
      "createdBy": "<account_id>",
      "created": 1541783367283,
      "id": "5643297d-8bfd-4bf5-b8b6-ef102efe737a",
      "fields": {
        "t": "ve",
        "h": "1541783367347"
      },
      "ownedBy": "<account_id>"
    },
    {
      "recordId": 1,
      "createdBy": "<account_id>",
      "created": 1541783823254,
      "id": "b2175873-b0f0-47b5-898d-630928ddeb68",
      "fields": {
        "t": "vew",
        "h": "1541783823321"
      },
      "ownedBy": "<account_id>"
    },
    {
      "recordId": 2,
      "createdBy": "<account_id>",
      "created": 1541783834097,
      "id": "a53198b0-32ed-48a3-b821-ffaafa1d0438",
      "fields": {
        "t": "vew2",
        "h": "1541783834159"
      },
      "ownedBy": "<account_id>"
    }
  ],
  "limit": 100,
  "startAt": 1,
  "total": 3
}

```

Example result returned by "Get"

/rest/api/v1/get/<TOKEN>/553943100/f/b2175873-b0f0-47b5-898d-630928ddeb68

```

{
  "recordId": 1,
  "createdBy": "<account_id>",
  "created": 1541783823254,
  "id": "b2175873-b0f0-47b5-898d-630928ddeb68",
  "fields": {
    "t": "vew",
    "h": "1541783823321"
  },
  "ownedBy": "<account_id>"
}

```