

# JSON-API of indexdatabase.de

Version 1.0 (Draft)

*Gunther Krauss*

*28 November 2017*

## Contents

<b>1 Introduction</b>	<b>1</b>
1.1 Status of the API . . . . .	1
1.2 Status of this document . . . . .	1
<b>2 Access</b>	<b>2</b>
<b>3 General Usage Scheme</b>	<b>2</b>
<b>4 Requests</b>	<b>3</b>
4.1 Available requests . . . . .	3
4.2 Chaining requests . . . . .	3
<b>5 Options</b>	<b>3</b>
<b>6 Result</b>	<b>4</b>
6.1 Result structure . . . . .	4
6.2 Example result . . . . .	4
<b>7 Error handling</b>	<b>6</b>
<b>8 Acknowledgements</b>	<b>6</b>

## 1 Introduction

This document describes the API<sup>1</sup> to retrieve data from the website [www.indexdatabase.de](http://www.indexdatabase.de) in an automated and machine readable way.

### 1.1 Status of the API

API is in beta stadium. Structure and naming of requests will not change in Version 1.x (unless there are major bugs/problems). Additional functionalities might be added.

### 1.2 Status of this document

This is a draft document. The information given here may not reflect the latest status of the API and all of its functionalities.

---

<sup>1</sup>Application Programming Interface

## 2 Access

Access to the IDB via the API is given to clients after registering and accepting the general terms and condition or signing a special agreement with IDB.

Please contact us by email [contact@indexdatabase.de](mailto:contact@indexdatabase.de) if you want to use the JSON API.

## 3 General Usage Scheme

The data is queried via REST<sup>2</sup> (by making HTTP-GET-Requests to specific urls).

The general URL-Scheme is:

```
https://www.indexdatabase.de/api/v<#.##>/<request>/?<options>
```

```
https://www.indexdatabase.de/api/v<#.##>/<request>/<param>/?<options>
```

```
https://www.indexdatabase.de/api/v<#.##>/<request>/<param>/<request>/<param>/?<options>
```

```
https://www.indexdatabase.de/api/v<#.##>/<request>/<param>/<request>/<param>/<request>/<param>/?<options>
```

where

- <#.##> is the version number of the API as a decimal number. Currently supported versions: 1.0
- <request>, e.g. indices, sensors, applications
- <param> is a parameter (in most cases an id) that is supplied to the requests.
- <options> a list of required or optional key=value pairs, separated by &
  - required keys: user, access\_token (for authentication)
  - optional: callback - name of a callback function to return the result object wrapped in the callback function as JSONP
  - optional, depending on request: page ...

Examples:

- Get first 20 indices:  
`https://www.indexdatabase.de/api/v1.0/indices/?user=foo&access_token=bar`
- Get indices 21 to 40:  
`https://www.indexdatabase.de/api/v1.0/indices/?user=foo&access_token=bar&page=2`
- Get index with id 17  
`https://www.indexdatabase.de/api/v1.0/indices/17/?user=foo&access_token=bar`
- Get indices for sensor with id 5  
`https://www.indexdatabase.de/api/v1.0/sensors/5/indices/?user=foo&access_token=bar`

The result is delivered as a text in JSON<sup>3</sup> format.

There is a maximum of 20 items returned per request. In order to retrieve all results of a request you have to repeat the request with incremented page.

The API offers read-only access via GET methods. POST, DELETE, ... calls to change data on server are not supported.

---

<sup>2</sup>Representational state transfer

<sup>3</sup>JavaScript Object Notation

## 4 Requests

### 4.1 Available requests

Request	Param	Options	Description
indices	-	page	Get 20 indices for page <sup>4</sup>
indices	<id>	-	Get index for index id <id>
sensors	-	page	Get 20 sensors for page page
sensors	<id>	-	Get sensor for index id <id>
applications	-	page	Get 20 applications for page
applications	<id>	.	Get applications for application id <id>
indices_by_name	-	search	Get indices that match search
sensors_by_name	-	search	Get sensors that match search

### 4.2 Chaining requests

To get e.g. the indices for sensor with id 12 one has to chain the requests:

```
/sensors/12/indices/
```

To get the sensors that are suitable to calculate the index with id 43 one has to use:

```
/indices/43/sensors/
```

Indices for application with id 15 and sensor 12 are retrieved via: `/applications/15/sensors/12/indices/`

Valid Request Chain	Description
indices > sensors	Get sensor(s) for specific index
indices > applications	Get applications(s) for specific index
sensors > indices	Get index/indices for specific sensor
sensors > applications	Get applications(s) for specific sensor
applications > indices	Get index/indices for specific application
applications > sensors	Get sensor(s) for specific application
applications > sensors > indices	Get index/indices for specific application and sensor

Other combinations are not supported (yet?) as they are meaningless or give identical results as one of the above.

## 5 Options

Key	Value	Required	Description
user	username	yes	Name the user is registered with
access_token	string	yes	A (quasi secret) token to authenticate with
page	number	optional	To get the entries $1+20*(page-1)$ to $20*(page)$
callback	function name	optional	Triggers javascript output and wraps result into callback function
context	string	optional	User defined string, useful with callbacks

<sup>4</sup>items from  $1+20*(page-1)$  to  $20*page$

## 6 Result

### 6.1 Result structure

The result is a string representing a JSON object with the fields.

- `apiVersion` - API version
- `status` - HTTP-Status code (200 if everything went well)
- `meta` - Metadata, including
  - `copyright`
  - `url` to indexdatabase's homepage
  - `date of creation`
- `data`
  - `kind` - the type of item (one of `indices`, `sensors`, `sensors_full`, `applications`, `indices_for_sensor`, `sensors_for_index`)
  - paging information as `currentItemCount`, `totalItems`, `itemsPerPage`, `startIndex`, `pageIndex`, `totalPages`<sup>5</sup>
  - `items` - array of items, possibly empty if request gives no result (e.g. wrong id or page too high)

The general type of the returned items is determined by the last (rightmost) request keyword, e.g. `/indices/` and `/sensors/12/indices/` return indices, whereas `/sensors/12/applications/` returns applications.

Notice that the preceding request keywords may alter the items' attributes. `/sensors/12/indices/` e.g. returns items, that contain additional information as the sensor specific formula and the names of sensor's bands used.

The structure of an item depends on it's type. Common fields are

- `id`
- `name`
- `url`

Additional fields depending on type:

- `indices`: `formula`, `abbrev`, `variant`, `variabledefinition`, `variableexplanation`
- `sensors`: `longname`, `numberofbands`, `infos on resolution etc.`
- `sensors_full`: `fields of sensor + additional array of bands`
- `sensors_for_index`: `formula - sensor specific formula`
- `indices_for_sensor`: `formula - sensor specific formula, generalformula,used_bands`

Fields of `bands/used_bands` items:

- `id`
- `bandnumber`
- `bandname`
- `start_nm`, `middle_nm`, `end_nm`

### 6.2 Example result

```
{  
  "apiVersion": 1.0,  
  "status": 200,  
  "meta": {
```

---

<sup>5</sup>see e.g. Google JSON Style Guide

```

    "copyright": "The Index Database - IDB",
    "url": "https://www.indexdatabase.de",
    "date": "2017-11-09CET01:53:35"
  },
  "data": {
    "kind": "indices_for_sensor",
    "currentItemCount": 20,
    "totalItems": 113,
    "itemsPerPage": 20,
    "startIndex": 1,
    "pageIndex": 1,
    "totalPages": 6,
    "sensor_id": 2,
    "items": [
      {
        "id": 574,
        "name": "Ashburn Vegetation Index ",
        "generalformula": "2.0*[800:1100] - [600:700]",
        "abbrev": "AVI",
        "variabledefinition": "",
        "variableexplanation": "",
        "wavelengths": "600:700,800:1100",
        "formula": "2.0 * [W18] - [W10]",
        "isderived": 1,
        "url": "https://www.indexdatabase.de/db/si-single.php?sensor_id=2&rsindex_id=574",
        "used_bands": [
          {
            "id": 112,
            "bandnumber": 10,
            "bandname": "W10",
            "start_nm": 646,
            "middle_nm": 651,
            "end_nm": 656
          },
          {
            "id": 120,
            "bandnumber": 18,
            "bandname": "W18",
            "start_nm": 1003,
            "middle_nm": 1019,
            "end_nm": 1036
          }
        ]
      },
      {
        "id": 4,
        "name": "Atmospherically Resistant Vegetation Index ",
        "generalformula": "(NIR - RED - y * (RED - BLUE)) / (NIR + RED - y*(RED-BLUE))",
        "abbrev": "ARVI",
        ...
      }
    ]
  }

```

## 7 Error handling

On error, the JSON Object has no data field but includes an error field with code and message. Additionally the status is set to an appropriate value different from 200.

Possible errors:

- Malformed url (e.g. no version string)
- Wrong version number
- Request does not exist
- User is not authenticated to submit the request (e.g. when user and access\_token are missing or wrong)
- The request is missing parameters, e.g. sensors\_by\_name with no search given.

## 8 Acknowledgements

We thank



for supporting the implementation of JSON-API. Check visualisation of indices on Sentinel Playground.