

Publisher Webservice

- [Project Management via the V6 REST-API](#)
- [Publisher Management via the V6 REST-API](#)

Project Management via the V6 REST-API

Introduction

The V6 REST-API provides administrators with the ability to manage projects efficiently. Using the `/Projects` endpoint, projects can be created, retrieved, updated, and deleted. Additionally, projects can be filtered using various parameters.

Endpoint

- **For Admins:** `https://SUBDOMAIN.de/ws/V6/admin/JSON/Projects`

Parameters & Filters

Required Parameters (for POST and PUT)

Parameter	Type	Description
<code>title</code>	String	Title of the project (Mandatory)
<code>status</code>	Integer	Status of the project (Mandatory)
<code>url</code>	String	URL of the project (Mandatory)

Filter Options (for GET)

Filtering can be applied to any column in the `publisher.projects` table, such as:

- `id`
- `status`
- `publisher_id`
- `projecttype`
- `title`
- `url`
- `channel_id`
- `hidden`

Example of a GET request with filters:

Bash



```
curl -X GET -H "Content-Type: application/json" -H "X-Network-ID: -1" -H "X-Auth-Token:
ADMIN_APIUSER_TOKEN" -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID"
"https://SUBDOMAIN.de/ws/V6/admin/JSON/Projects?status=1&publisher_id=10"
```

CRUD Operations

GET: Retrieve Projects

Retrieve a list of projects or a specific project by its `id`.

Bash



```
curl -X GET -H "Content-Type: application/json" -H "X-Network-ID: -1" -H "X-Auth-Token:
ADMIN_APIUSER_TOKEN" -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID"
"https://SUBDOMAIN.de/ws/V6/admin/JSON/Projects?id=5"
```

POST: Create a New Project

Bash



```
curl -X POST -H "Content-Type: application/json" -H "X-Network-ID: -1" -H "X-Auth-Token:
ADMIN_APIUSER_TOKEN" -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID" -d '{
  "title": "New Project",
  "status": 1,
  "url": "https://example.com"
}' "https://SUBDOMAIN.de/ws/V6/admin/JSON/Projects"
```

PUT: Update an Existing Project

Bash



```
curl -X PUT -H "Content-Type: application/json" -H "X-Network-ID: -1" -H "X-Auth-Token:
ADMIN_APIUSER_TOKEN" -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID" -d '{
  "id": 5,
  "title": "Modified Project",
  "status": 2,
  "url": "https://updated-example.com"
}' "https://SUBDOMAIN.de/ws/V6/admin/JSON/Projects"
```

DELETE: Delete a Project

Delete a project by its `id`.

Bash



```
curl -X DELETE -H "Content-Type: application/json" -H "X-Network-ID: -1" -H "X-Auth-Token:
ADMIN_APIUSER_TOKEN" -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID"
"https://SUBDOMAIN.de/ws/V6/admin/JSON/Projects?id=5"
```

Responses

Successful Response

JSON



```
{
  "id": 5,
  "status": 1,
  "publisher_id": 10,
  "projecttype": "page",
  "title": "Project Example",
```

```
"description": "Description of the project",
"url": "https://example.com",
"channel_id": 2,
"project_identifier_string": "abc123",
"reach": "1000",
"statistic_type": "internal",
"external_sources_entity_id": "-1",
"trackingtemplate": "template",
"kpi_whitelist": "kpi1,kpi2",
"hidden": false,
"login_type": "pub",
"exclude_from_salary": 0,
"default_project": false,
"insert_timestamp": "2025-01-27T12:00:00"
}
```

Bad Request

JSON

```
{
  "error": "Missing required field: title"
}
```

? Error Handling Notes

- If mandatory fields (`title`, `status`, `url`) are missing, an appropriate error message will be returned.
- An invalid or non-existent `id` during a **GET**, **PUT**, or **DELETE** request will result in an empty response or an error message.

Publisher Management via the V6 REST-API

Introduction

The V6 REST-API allows administrators to create publishers and retrieve their details. This functionality is currently limited to **CREATE** and **READ** operations.

Endpoint

- **For Admins:** `https://SUBDOMAIN.de/ws/V6/admin/JSON/Publisher`

CRUD Operations

POST: Create Publisher (CREATE)

Creates a new publisher and inserts the corresponding data into the `global.logins` and `global.login_settings` tables.

Required Parameters:

Parameter	Type	Description
<code>email</code>	String	Mandatory, max. 255 characters
<code>salutation</code>	String	Mandatory, values: <code>mr</code> or <code>mrs</code>
<code>prename</code>	String	Mandatory, max. 255 characters
<code>surname</code>	String	Mandatory, max. 255 characters

- **Optional Parameters (Excerpt):** `company`, `street`, `zip`, `city`, `telephone`, `country` (ISO 3166 ALPHA-3), `billing_sepa_iban`, `companytype` (`un` for company or `priv` for private), `language_interface` (Default: `DEU`), etc.

Example Request:

Bash
❏ ❏

```
curl -X POST -H "Content-Type: application/json" -H "X-Network-ID: -1" -H "X-Auth-Token: ADMIN_APIUSER_TOKEN" -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID" -d '{
  "email": "max.mustermann@example.com",
  "salutation": "mr",
  "prename": "Max",
  "surname": "Mustermann",
  "company": "Musterfirma",
  "street": "Musterstraße 1",
  "zip": "12345",
  "city": "Musterstadt",
  "country": "DEU",
  "billing_sepa_owner": "Max Mustermann",
  "billing_sepa_iban": "DE12345678901234567890",
  "billing_sepa_bic": "GENODEF1M01",
  "language_interface": "DEU",
  "billing_mode": 1,
  "billing_media": 1,
  "country_billing": "DEU",
  "country_publisher": "DEU",
  "billing_limit": 50
}' "https://SUBDOMAIN.de/ws/V6/admin/JSON/Publisher"
```

Expected Response:

JSON

■ ■

```
{
  "id": 123,
  "name": "Max Mustermann",
  "email": "max.mustermann@example.com"
}
```

GET: Retrieve Publisher (READ)

Returns the details of a specific publisher or a list of publishers.

- **Filter Options:** All fields within the table can be used as filters (e.g., `email`, `status`, `prename`, `surname`).

Example Request (Specific ID):

Bash



```
curl -X GET -H "Content-Type: application/json" -H "X-Network-ID: -1" -H "X-Auth-Token: ADMIN_APIUSER_TOKEN" -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID" "https://SUBDOMAIN.de/ws/V6/admin/JSON/Publisher?id=123"
```

Example Request with Filters:

Bash



```
curl -X GET -H "Content-Type: application/json" -H "X-Network-ID: -1" -H "X-Auth-Token: ADMIN_APIUSER_TOKEN" -H "X-Auth-ID: ADMIN_APIUSER_LOGIN_ID" "https://SUBDOMAIN.de/ws/V6/admin/JSON/Publisher?status=1&prename=Max"
```

Error Handling

Missing Required Fields (POST):

JSON



```
{
  "error": "Missing required field: email"
}
```

Publisher Not Found (GET):

JSON



```
{  
  "error": "Publisher not found"  
}
```