# FlexPoint Participant API Integration Guide

Version
5.1.11

## Overview

The DERMS exposes a RESTful JSON API over the public internet. This API allows demand response participants to manage their own sites and registrations, and to respond to events. This document's purpose is to detail the design and usage of this API.

## Design

The DERMS Participant API is designed to conform to the JSON: API Specification

> *From https://jsonapi.org/*
>
> *JSON: API is a specification for how a client should request that resources be fetched or modified, and how a server should respond to those requests. JSON: API can also be easily extended with profiles.*
>
> *JSON: API is designed to minimize both the number of requests and the amount of data transmitted between clients and servers. This efficiency is achieved without compromising readability, flexibility, or discoverability.*
>
> *JSON: API requires use of the JSON:API media type (application/vnd.api+json) for exchanging data.*

### Security

The API is only accessible over HTTPS, on port 443. All requests require user authentication (except for those specifically related to credential management). You must have an existing DERMS user account to be able to use the API.

#### Authentication Token

A request is authenticated by providing a current authentication token. Note: an **authentication token** lasts exactly 5 minutes, after this period has elapsed you must exchange your **refresh token** for a fresh authentication token.
You may also refresh your **authentication token** at any point within the 5 minute window to ensure uninterrupted access. See Authentication for examples on how to generate tokens.

### Access

The base URL for the API is

```
https://app.flexpoint.co.nz/api
```

The base URL for the UAT environment is

```
https://uat.flexpoint.co.nz/api
```

There are no IP access restrictions in place on the API. You can verify you can access the API by initiating a connection to app.flexpoint.co.nz/api or uat.flexpoint.co.nz/api on port 443.

E.g.

```
telnet app.flexpoint.co.nz 443
Trying 104.210.67.137...
Connected to drmsaueprodapp.australiaeast.cloudapp.azure.com.
Escape character is '^]'.
^CConnection closed by foreign host.
```

## Authentication

- Authentication is handled by a combination of an authentication token and a refresh token. The authentication token is required to authenticate all API requests.
- The refresh token can be used to generate a new authentication token when the current one expires.

Note, the refresh token also has an expiry but it has a much longer expiry window. Once the refresh token expires a new one must be generated by signing in again.

To generate a new pair of refresh and authentication tokens:

| **Endpoint** | **Method** | **Authentication** |
|---|---|---|
| /tokens | POST | none |
| **Payload** | **Response** | |
| {<br>  "data": {<br>    "attributes": {<br>      "email": "john.smith@example.com",<br>      "password": "Sup3rS3cur3!"<br>    }<br>  }<br>} | {<br>  "data": {<br>    "type": "credentials",<br>    "attributes": {<br>      "auth": "eyJhbGciOiJIUzI1NiJ9...",<br>      "refresh": "eyJhbGciOiJIUzI1NiJ9....",<br>      "needChangePassword": false<br>    }<br>  }<br>} | |

Once you have retrieved an authentication token you must use this token to authenticate all subsequent requests. The token is passed to the API via the **Authorization** header.
E.g.

```
$ curl -H 'Authorization: eyJhbGciOiJIUzI1NiJ9…' -H 'Content-Type:
application/json' https://app.flexpoint.co.nz/api/sites
```

### Token Structure

This application uses **JWT** tokens for authentication. A JWT is a combination of a header, an encoded data structure, and a signature that verifies its origin.
The DERMS application will only allow access to tokens that have a valid signature (and must therefore have been generated by the DERMS itself),and have a future expiry time.

### What's inside the JWT?

The JWT identifies the currently authenticated user,vand contains information about this user as well as the organisation they belong to. To inspect the contents of the JWT it needs to be decoded.
To read more about JWT or to decode one online visit [https://jwt.io](https://jwt.io).

The structure of a JWT token issued by the DERMS looks like this:

```
{
  "user": {
    "data": {
      "id": "88",
      "type": "users",
      "attributes": {
        "name": "Demo User",
        "email":
        "user@flexpoint.co.nz",
        "status": "active",
        "active": true,
        "combinedSystemPermissions": {
          //User system permissions
        }
      },
      "relationships": {
        "organisation": {
          "data": {"type": "organisations","id": "1"}
        },
        "roles": {
          "data": [
            {"type": "userRoles","id": "10"},
            {"type": "userRoles","id": "1196"}
          ]
        }
      }
    },
    "included": [
      {
        "id": "1",
        "type": "organisations",
        "attributes": { "name": "Transpower" }
      },
      {
        "id": "10",
        "type": "userRoles",
        "attributes": {
          "name": "System Admin"
        },
        "relationships": {...}
      },
      {
        "id": "1196",
        "type": "userRoles",
        "attributes": {
          "name": "Admin"
        },
        "relationships": {...}
      }
    ],
    "jsonapi": {
      "version": "1.0"
    }
  },
  "type": "auth", "exp":
  1562207278,
  "generated_at": 1562206978
}
```

You can use the token to identify the following information for the authenticated user:

- The name and ID of the user's parent organisation
- The name of any roles the user holds
- The users status, name and email address

### Token expiration

The token expiration is encoded as a unix timestamp (measured as the number of seconds since January 1st, 1970 at UTC). Once the expiry has elapsed the token is no longer valid.

Refresh tokens have a relatively short life-span by default (30 minutes), but the **rememberMe** option can be used to generate a refresh token with a longer life-span (30 days). Note that to avoid a user session being abruptly interrupted there is some leeway built in to refresh tokens. This works as follows:

- A refresh token can be **indefinitely** refreshed so long as the token is exchanged for a new token pair **at least once within 5 minutes of the token being generated**. This ensures that a user actively using the DERMS is never interrupted mid work-flow, but they will be automatically logged out after 5 minutes of inactivity **if** their refresh token has expired.

### Token invalidation

Tokens are invalided when they expire, **or** if the user has initiated a password reset since the token was generated.

## Endpoints

### POST/tokens

Sign in as a user. The response will return two JWT tokens.

1. **auth.** This token should be passed in the Authorisation header for all requests that require authentication. E.g. `$ curl -H "Authorization: [auth]" https://app.flexpoint.co.nz/api/..."`
2. **refresh.** This token should be used to generate a new auth token when your current one expires. Use the PUT/tokens endpoint for this.

The **rememberMe** option generates a long-lived refresh token that doesn't need to be regenerated for 30 days. If this option is not provided the default refresh token will last for 30 minutes.

| **Endpoint** | **Method** | **Authentication** |
|---|---|---|
| /tokens | POST | none |
| **Payload** | **Response** | |
| `{`<br>`  "data": {`<br>`    "attributes": {`<br>`      "email": "john.smith@example.com",`<br>`      "password": "Sup3rS3cur3!",` | `{`<br>`  "data": {`<br>`    "type": "credentials",`<br>`    "attributes": {`<br>`      "auth": "eyJhbGciOiJIUzI1NiJ9...",` | |

| | |
|---|---|
| "rememberMe": true<br>}<br>}<br>} | "refresh": "eyJhbGciOiJIUzI1NiJ9....",<br>"needChangePassword": false<br>}<br>}<br>} |

### PUT /tokens

This endpoint will generate a fresh auth token, given a valid refresh token. It requires you to use the **refresh** token in place of the **auth** token in the **Authorization** header.

| **Endpoint** | **Method** | **Authentication** |
|---|---|---|
| /tokens | PUT | **Use Refresh token instead of Auth token in the authorization header** |
| **Payload** | **Response** | |
| | {<br>"data": {<br>"type": "credentials",<br>"attributes": {<br>"auth": "eyJhbGciOiJIUzI1NiJ9...",<br>"refresh": "eyJhbGciOiJIUzI1NiJ9....",<br>"needChangePassword": false<br>}<br>}<br>} | |

## Authorization

All authenticated endpoints require authorization. Upon invoking an endpoint the active user is evaluated against one or more required permissions, if the required permissions are not present the request is rejected.

If authorization is rejected the following error is returned

| | | |
|---|---|---|
| 401 | ERR_UNAUTHORIZED | Not Authorised |

## Request format & encoding

All requests should have a content type of `application/json`. All responses will also have this content type. See RequestHeaders**.**

All URLs including query strings should be URL encoded. See: https://www.w3schools.com/tags/ref_urlencode.asp for more information.

## Request Headers

1. `Content-type: application/json`
   This header is expected on all requests to identify the content type of the request. The response will also have a `Content-type: application/json` header set.
2. `Authorization: [AUTH TOKEN]`
   This header is expected on all requests that require authentication. Here **AUTH_TOKEN** is expected to be a current authentication token, requested by logging in via the `/tokens` endpoint.
3. `X-Authorization: [AUTH TOKEN]`
   This is an alternate form of the authentication header that serves the same purpose as the `Authorization` header. You can use this header if you are sending API requests from within a network that strips or rewrites `Authorization` headers.
4. `Accept-Encoding: gzip, deflate, br`. These optional headers indicate the API client supports one of the listed compression schemes. In this scenario the API results will be compressed. JSON:API responses lend themselves very well to compression and achieve a very high compression ratio.

## API Versioning

The participant API is a versioned API. If you make requests without specifying a version, you will by default invoke the latest version of the API (currently v1). To use a specific version of the API an optional **Accept** header can be used.

E.g.

```
$ curl -H 'Accept: application/flexpoint; version=1' ...
```

**Note: At the time of writing, only a single version of API routes exists:** `version=1`

## Request Structure

### Sorting

When an endpoint supports sorting of results, the sort order is specified with an `order` query parameter. The sort order for each sort field will be ascending unless it is prefixed with a minus.

E.g.

```
$ curl 'https://app.flexpoint.co.nz/api/child_events?order=-
start' -H "Content-Type: application/json" -H "Authorization:
eyJhb...."
```

*Sort child events by event **start** (descending)*

### Pagination

All index endpoints (e.g. GET /sites or GET /registrations) are paginated. The API uses a default page size of 15 items per page. A consumer can optionally specify an alternate page-size up to hard limit of 250 items per page.

E.g.

```
https://app.flexpoint.co.nz/api/registrations?page[number]=2
```

*Request the second page of the collection*

```
https://app.flexpoint.co.nz/api/registrations?page[number]=2&page[size]=50
```

*Request the second page of the collection, and set the page size to 50 items per page*

### Filtering

When an endpoint supports filtering of results, the sort order is specified with a `filter` query parameter. Multiple filters can be combined into a single request.

E.g.

```
https://app.flexpoint.co.nz/api/registrations?filter[status]=active&filter[fro
m]=20
19-06-30T22:35:37.327Z

https://app.flexpoint.co.nz/api/sites?filter[active]=false&filter[name]=My%
20Specia l%20Site
```

## Relationships

The API supports requesting related entities alongside the root entity in a single request. This allows you to request nested data within a single request, saving the need for multiple network round trips to retrieve data.

When an endpoint supports requesting of relationships, the included relationships are specified with the **relationships** query parameter. Multiple relationships can be included as a comma separated string. Nested relationships are specified using the '.' separator.

E.g.

```
$ curl
'https://app.flexpoint.co.nz/api/child_events?include=offer,registration.s
ites' -H "Content-Type: application/json" -H "Authorization: eyJhb...."
```

*Request child events and also include information about*

*I - The offer associated with the child event*

*2- The registration associated with the child event*

*3- The sites beneath the associated registration (note the use of the '.' for the nested relationship)*

## Sparse Fieldsets

The API supports requesting a sub-set of all supported fields. This can be used to reduce request size, and craft targeted responses containing only what the client requires.

When an endpoint supports requesting of sparse fieldsets, the included fields are specified with the **fields** query parameter. The fields parameter can be used to filter fields on both the root resource, and any requested relationships (see Relationships). Multiple fields are included as a comma separated string.

E.g.

```
$ curl
'https://app.flexpoint.co.nz/api/child_events?fields[childEvents]=targetKw,mwh
Price
,registration&fields[registration]=name&include=registration.sites' -H
"Content-Type: application/json" -H "Authorization: eyJhb...."
```

*Request only the **targetKw**, **mwhPrice** and **registration** for each child event*

*Request only the name for each included registration (note the presence of the **include** parameter, to specify this relationship must be present, see also: Relationships)*

## Response Structure

The response of an API request will be a JSON API document. *For the section of the JSON API specification on document structure look here.*

A response will always:

- Consist of a single top-level JSON object (not an array)
- Contain a **data** member (if not an error or meta with no resource)
- Contain an **errors** member (if is an error)
- May optionally contain a **meta** member to report non-standard meta-information.
- Data may either be a single resource, or an array of resources

```
{
    "data": [...resource objects],
    "meta": null
}
```

*Example structure for a success response*

```
{
    "error": [...error objects],
    "meta": null
}
```

*Example structure for an error response*

A document may optionally contain multiple resource types, if additional relationships are requested, as specified in: **Relationships**

```
{
    "data": {
        "id": 3485,
        "type" : "sites",
        "attributes" : {
            "name": "Demo Site",
            "address": "1 main street",
        },
        "relationships": {
           "registration" : {
             "data" : {
               "type": "registrations",
               "id": 2
             }
           }
        }
    },
    "included": [
       {
         "id": 2,
         "type": "registrations",
         "attributes": {
            "name": "Demo Registration",
          }
       }
    ]
}
```

*A response containing a single resource object as the* ***data*** *member, and included relationships (a registration). Note that by including additional resources separately and expressing relationships through linkage, the document never contains duplicate entities, and can express complex relationships including cyclic relationships.*

See also https://jsonapi.org/format/#document-compound-documents

### Resource Object

A successful response will contain one or more resource objects. (e.g. a site, registration etc).

A resource object will **always** have the following members:

- `id` – a unique identifier for this resource
- `type` – a string (plural) describing the type of the resource object

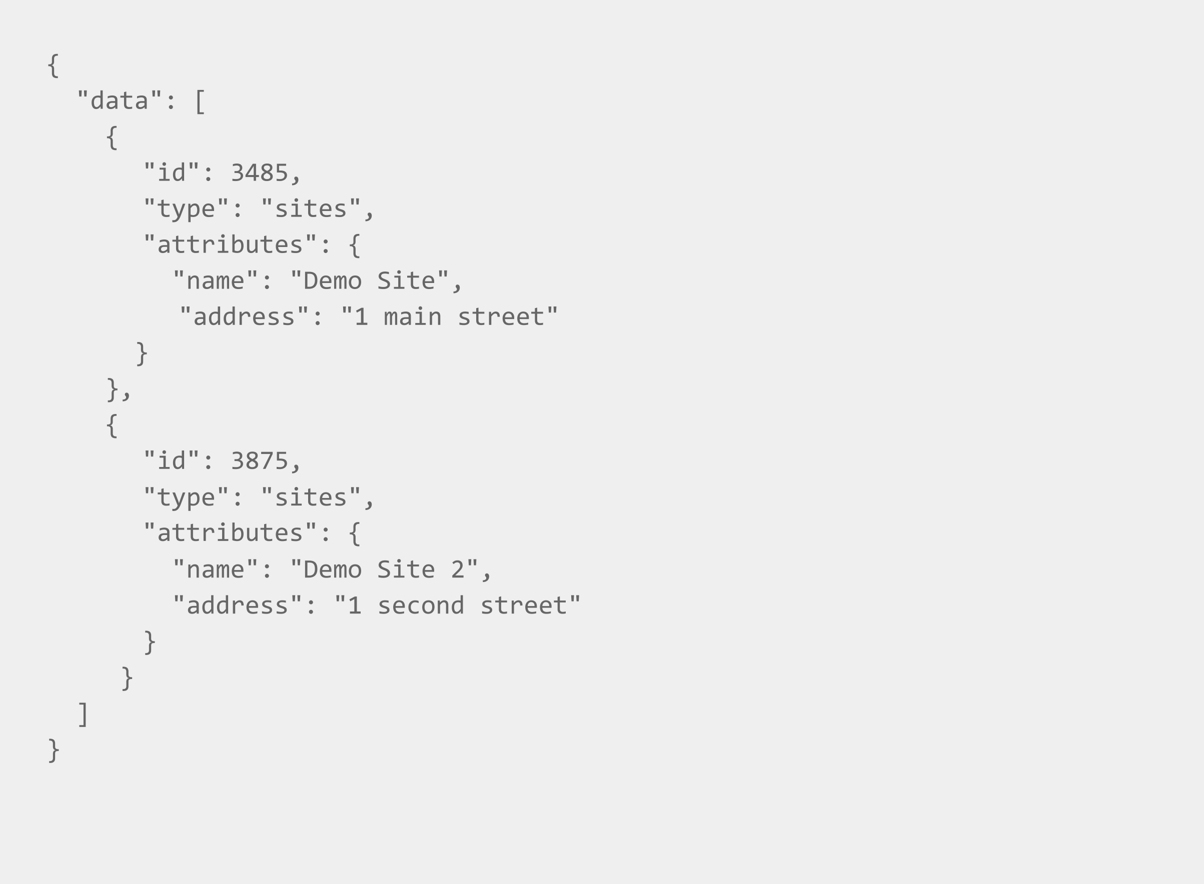
It may optionally have the following members:

- `attributes`: an attributes object (name, value pairs) representing some of the resource's data.
- `relationships`: a relationships object (name, value pairs) describing relationships between the resource and other JSON:API resources.
- `meta`: a metaobject containing non-standard meta-information about a resource that can not be represented as an attribute or relationship.

E.g.

```
{
    "data": {
        "id": 3485,
        "type" : "sites",
        "attributes" : {
            "name": "Demo Site",
            "address": "1 main street",
        }
    }
}
```

*A response containing a single resource object as the **data** member*



```
{
  "data": [
    {
      "id": 3485,
      "type": "sites",
      "attributes": {
        "name": "Demo Site",
         "address": "1 main street"
      }
    },
    {
      "id": 3875,
      "type": "sites",
      "attributes": {
        "name": "Demo Site 2",
        "address": "1 second street"
      }
     }
  ]
}
```

*A response containing a collection of resource objects as the **data** member*

### Error Object

An error object has the following members:

- Status (an error status number)
- Code (a readable error short code)
- Title
- Message

An error response always returns an array of errors in the top-level **errors** member.
By convention the API will always return the primary/most significant error at index 0.

E.g.

```
{
  "errors": [
    {
      "status": 422,
      "code": "ERR_INVALID_CREDENTIALS",
      "title": "Invalid Credentials",
      "message": "Invalid Account Credentials"
    }
  ]
}
```

## API Errors

The following table contains errors that may be reported by the API:

| Status | Code | Title |
|---|---|---|
| 403 | ERR_NEED_OLD_PASSWORD | Current password required |
| 401 | ERR_NOT_AUTHENTICATED | Not Authenticated |
| 406 | ERR_BAD_REQUEST | Bad Request |
| 409 | ERR_DELETE_RESTRICTED | Delete Restricted |
| 406 | ERR_OVERLAPPING_EVENTS | Conflicting events |
| 406 | ERR_LOCK_FAILED | Application lock failure |
| 401 | ERR_INACTIVE_USER | Inactive user account |
| 404 | ERR_ROUTE_NOT_FOUND | Route Not Found |
| 409 | ERRO_INSUFFICIENT_BASELINE_DAYS | Not enough valid baseline days |
| 401 | ERR_TOKEN_EXPIRED | Token Expired |
| 422 | ERR_INVALID_CREDENTIALS | Invalid Credentials |
| 401 | ERR_UNAUTHORIZED | Not Authorised |
| 404 | ERR_NOT_FOUND | Not Found |
| 500 | ERR_UNEXPECTED | Unexpected Error |
| 406 | FAILED_METER_DATA_IMPORT | Invalid meter data file |
| 422 | ERR_BAD_REQUEST | Unprocessable entity |
| 406 | ERR_INVALID_RECORD | Error |
| 400 | ERR_MALFORMED_CSV | Malformed CSV |
| 404 | ERR_MISSING_METER_DATA | Missing meter data |
| 406 | ERR_MISSING_PARAM | Missing Parameter |

## API Limits

The API enforces usage limits to ensure a consistent level of service for all consumers. The following usage limits apply:

- A hard limit of no more than 20 requests per second is applied.
- Rate limiting **may** be applied to any IP address exceeding 10 requests per second over an extended period
- Session/token related endpoints are limited to no more than 6 requests per 60s

If a rate limit is applied, the API will return a 429 status code. Upon receiving a 429 a client should follow randomised back-off procedure e.g. exponentialback-off before attempting to try the request again.

# Use Cases

These use cases are example scenarios describing how you can combine a variety of the below endpoints to achieve a specific outcome.

The examples in these use cases can be executed using the **curl** utility from the command line. These exact same use-cases could also be enacted using application code, or an API testing tool like Postman.

> ℹ **It is important** to remember to use the **Content-type: application/json** header for any requests that contain a JSON payload (i.e. POST and PUT requests)

## Creating a new site

To create a new site, you must know the following: ICP, Meter ID, Address, Flow Direction, GXP, Distributor, Meter Owner, Retailer, which verification method the site will use and your organisation ID.

For example a user might follow these steps:

1. Sign and retrieve an auth token

```
$ curl -H 'Content-Type: application/json' -H 'Authorization: [auth]'  -XPOST
https://app.flexpoint.co.nz/tokens -d '{"data":
{"attributes":{"email":"user@example.com", "password":"Sup3rS3cret"}}'

# Response
{"data":{"type":"credentials","attributes":{"auth":"eyJhb...","refresh":"eyJhbG...","needChange
Password":false}}}
```

2. Take the **"auth"** token from this response and decode it to discover my organisation ID. Read more here to find out about where this information can be found: What's insidetheJWT?

3. Discover the ID for my GXP (HAY2201), Retailer (Meridian), Distributor (Wellington Electricity) and Metering Equipment Provider (Advanced Metering Services)

*Look for Meridian*

```
$ curl -H 'Authorization: [auth]'
"https://app.flexpoint.co.nz/api/organisations/retailers?filter[nam
e]=Meri"

# Response
{
   "data": [
       {
           "id": "5",
           "type": "organisations",
           "attributes": {
               "name": "Meridian Energy"
           }
       }
   ],
   "meta": {
       "totalPages": 1,
       "count": 2,
       "page": 1
   },
   "jsonapi": {
       "version": "1.0"
   }
}
```

*Look for Wellington Electricity*

```
$ curl -H 'Authorization: [auth]'
"https://app.flexpoint.co.nz/api/organisations/distributors?filter[name]=
Welling"

# Response
{
    "data": [
        {
            "id": "7",
            "type": "organisations",
            "attributes": {
                "name": "Wellington Electricity"
            }
        }
    ],
    "meta": {
        "totalPages": 1,
        "count": 2,
        "page": 1
    },
    "jsonapi": {
        "version": "1.0"
    }
}
```

*Look for Advanced Metering Services*

```
$ curl -H 'Authorization: [auth]'
"https://app.flexpoint.co.nz/api/organisations/meter_owners?filter[name]=A
dvanced"

# Response
{
    "data": [
        {
            "id": "9",
            "type": "organisations",
            "attributes": {
                "name": "Advanced Metering Services Limited"
            }
        }
    ],
    "meta": {
        "totalPages": 1,
        "count": 2,
        "page": 1
    },
    "jsonapi": {
        "version": "1.0"
    }
}
```

*Look for GXP HAY2201*

```
$ curl -H 'Authorization: [auth]'
"https://app.flexpoint.co.nz/api/gxps?filter[code]=HAY2201"

# Response
{
   "data": [
     {
       "id": "124",
       "type": "gxps",
       "attributes": {
         "code": "HAY2201"
       }
     }
   ],
   "meta": {
     "totalPages": 1,
     "count": 1,
     "page": 1
   },
   "jsonapi": {
     "version": "1.0"
   }
 }
```

*Look for the verification method with type 3-Day SAA*

```
$ curl -H 'Authorization: [auth]'
"https://app.flexpoint.co.nz//api/verification_methods?filter[name
]=3-Day"

# Response
{"data":[{"id":"1","type":"verificationMethods","attributes":{...}}],"jsonapi":{"vers
ion":"1.0"}}
```

4. Create the new site (In this example the organisation ID retrieved in step 2 is **5**). The IDs for the distributor, retailer, meterOwner and verificationMethod are retrieved in step 3.

```
$ curl -H 'Content-Type: application/json' -H 'Authorization: [auth]' -XPOST
"https://app.flexpoint.co.nz/api/organisations/5/sites" -d '
{
   "data": {
     "type": "sites",
     "attributes": {
       "status": "active",
       "name": "ACME Sawmill",
       "icpNumber": "8671784589NI73E",
       "meterId": "10807243",
       "address": "1 Main Street",
       "flowDirection": "X-I",
       "loads": { "Lighting":
         100,
         "Refrigeration": 100
       }
     },
     "relationships": {
       "gxp": {
         "data": {"id": "124","type": "gxps"}
       },
       "retailer": {
         "data": {"id": "5","type": "organisations"}
       },
       "distributor": {
         "data": {"id": "7","type": "organisations"}
       },
       "meterOwner": {
         "data": {"id": "9","type": "organisations"}
       },
       "verificationMethod": {
         "data": {"id": "1","type":"verificationMethods"}
       }
     }
   }
 }
 '

# Response
{"data":{"id":"11","type":"sites","attributes":{"name":"ACME
Sawmill","icpNumber":"8671784589NI73E","meterId":"10807243","...},"relationships":{"organisati
on":...},"jsonapi":{"version":"1.0"}}
```

## Enrolling sites into a programme

To enroll one or more sites into a programme you need to know the ID of the programme into which you would like to enroll. Once you have created a draft registration you can also:

- Submit the registration for review by an admin
- Query the registration to see its current status.

This use case assumes you have already created one or more sites under your registration, and know the IDs of these sites

1. Find the ID of the programme you are looking for

*Looking for the "Price Responsive Programme"*

```
$ curl -H 'Authorization: [auth]'
"https://app.flexpoint.co.nz/api/programmes?filter\[name\]=Price%20Responsive&filt
er[price
_responsive]=true"

# Response
{
   "data": [
    {
      "id": "1",
      "type": "priceResponsiveProgrammes",
      "attributes": {
        "name": "Price responsive programme",
        "startDate": "2018-07-02",
        "endDate": "2020-07-02",
        "minimumLeadTime": 120,
        "type": "PriceResponsiveProgramme",
        "tags": {},
        "deviceFilter": {},
        "active": true,
        "signalMappings": {},
        "requiresFixedPrice": true,
        "requiresAvailabilityFee": false,
        "requiresPrepurchasedHours": false,
        "allowsEstablishmentFee": null,
        "readOnly": false,
        "autoDR": false
      },
      "relationships": {
        "organisation": {
          "meta": {
            "included": false
          }
        },
        "availabilities": {
          "meta": {
            "included": false
          }
        }
      }
    }
  ],
  "meta": {
    "totalPages": 1,
    "count": 1,
    "page": 1
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

2. Construct the registration (This assumes you know your organisation ID and the IDs of the sites you want to enroll, see the **Creating a new site** use case to find our more about this)

```
$ curl -H 'Authorization: [auth]' -XPOST -H 'Content-Type: application/json'
"https://app.flexpoint.co.nz/api/organisations/5/registrations" -d '
{
   "data": {
     "attributes": {
       "name": "ACME Reg 4",
       "startDate": "2019-07-02",
       "endDate": "2020-07-02",
       "organisationId": "1",
       "initialEstablishmentFee": {
         "amount": 7.5,
         "date": "2019-07-02T20:49:32.821Z"
       },
       "finalEstablishmentFee": {
         "amount": 7.5,
         "date": "2020-07-02"
       },
       "indicativePrice": 50
     },
     "relationships": {
       "sites": {"data": [{"id": "11","type": "sites"}]},
       "programme": {"data": {"id": "1","type": "programmes"}}
     }
   }
 }

'

# Response
{"data":{"id":"14","type":"registrations","attributes":{"name":"ACME Reg
4","status":"submitted", ...},"relationships":{...}},"jsonapi":{"version":"1.0"}}
```

3. Once the draft registration looks correct, you can submit it for review

```
$ curl -H 'Authorization: [auth]' -XPOST -H 'Content-Type: application/json'
"https://app.flexpoint.co.nz/api/registrations/14/events" -H -d '
{
   "data": {
     "attributes": {
       "name": "submit"
     }
   }
 }
 '

# Response
{"data":{"id":"","type":"events","attributes":{"name":"submit","options":{}},"rela
tionships":{"target":{"meta":{"included":false}}}},"jsonapi":{"version":"1.0"}}
```

4. You can poll periodically to see the status of your registration

```
$ curl -H 'Authorization: [auth]' "https://app.flexpoint.co.nz/api/registrations/14"

# Response
{"data":{"id":"14","type":"registrations","attributes":{"name":"ACME Reg
4","status":"submitted", ...},"relationships":{...}},"jsonapi":{"version":"1.0"}}
```

## Responding to an event

You can poll child events periodically to see if there are any pending events.

1. If there is a child event in the offer_window_open status, you can respond to this offer.

```
$ curl -H 'Authorization: [auth]' "https://app.flexpoint.co.nz/api/registrations/14"
```

```
# Response
{
   "data": [
     {
       "id":"14",
       "type":"childEvents",
       "attributes": {
        "status": "offer_window_open
         ...
       }
     }
  ],
   "meta": {
       "totalPages": 0,
       "count": 0,
       "page": 1
   },
   "jsonapi": {
       "version": "1.0"
   }
}
```

2. Fetch the event details, including the corresponding registration, parent event and offer

```
$ curl -H 'Authorization: [auth]'
"https://app.flexpoint.co.nz/api/child_events/14?include=offer,parentEvent,registr
ation&fi
elds[childEvents]=status&fields[offers]=offerKw,offerPrice,status,mwhPrice&fields[
registrations]=name&fields[parentEvent]=start,end"

# Response
{
   "data": {
       "id": "1",
       "type": "childEvents",
       "attributes": {
           "status": "offer_window_open"
       }
   },
   "included": [
       {
           "id": "1",
           "type": "parentEvents",
           "attributes": {
               "start": "2019-07-04T20:00:00.000+12:00",
               "end": "2019-07-04T22:00:00.000+12:00",
               "status": "offer_window_open",
               "name": "Price responsive programme 04/07/2019 20:00-04/07/2019
22:00",
               "offerWindowStart": "2019-07-04T15:59:35.879+12:00",
               "offerWindowEnd": "2019-07-04T17:40:00.000+12:00"
           },
           "relationships": {
               "organisation": {
                   "meta": {
                       "included": false
                   }
               }
           }
       },
       {
           "id": "14",
           "type": "registrations",
           "attributes": {
               "name": "ACME Reg 4"
           }
       },
       {
           "id": "1",
           "type": "offers",
           "attributes": {
               "status": "pending",
               "offerKw": 200,
               "offerPrice": 50
           }
       }
   ],
   "jsonapi": {
       "version": "1.0"
   }
}
```

3. Respond to the offer by opting in or out and optionally updating the offer kW and Price (using the offer ID from above)

```
$ curl -XPUT  -H 'Authorization: [auth]'  -H 'Content-Type: application/json'
'https://app.flexpoint.co.nz/api/offers/1' -d '
{
  "data": { "attributes": {
       "status": "opted_in",
       "offerKw": 100,
       "offerPrice": 10
       }
  }
}

'

# Response
{
    "data": {
        "id": "1",
        "type": "offers",
        "attributes": {
            "status": "opted_in",
            "offerKw": 100,
            "nonAutoDRKw": null,
            "autoDRKw": 0,
            "offerPrice": 10,
            "targetKw": 200,
            "targetPrice": 50,
            "drLevel": null
        },
        "relationships": {...},
    "jsonapi": {
        "version": "1.0"
    }
}
```

4. Wait for the offer window to be closed, then check to see if the offer was accepted

```
$ curl -H 'Authorization: [auth]'
"https://app.flexpoint.co.nz/api/child_events/14?fields[childEvents]=status&fields
[offers]
=offerKw,offerPrice,status,mwhPrice&fields[registrations]=name&fields[parentEvent]
=start,end"

# Response
{
   "data": {
       "id": "1",
       "type": "childEvents",
       "attributes": {
           "status": "scheduled"
       }
   },
   "jsonapi": {
       "version": "1.0"
   }
}
```

## Troubleshooting

If you encounter difficulties executing any of these API requests or use-cases you can try:

1. Refer to the APIErrors table to see detect if the issue you are encountering is due to a data or misconfiguration issue
2. Verify that the API user you are using has sufficient permissions to execute the requests. If not you may need an administrative user to grant you permission.

# Services

## Sites

The **site** endpoints allow you to manage participant sites in the DERMS.

### Resource

#### Schema

```
{
  "data": {
    "id": "string",
    "type": "string",
    "attributes": {
        "name": "string",
        "icpNumber": "string",
        "meterId": "string",
        "consumerAuthorisationCode": "string",
        "consumerNo": "string",
        "customerName": "string",
        "address": "string",
        "status": "string",
        "kwAmount": 0,
        "flowDirection": "string",
        "tags": "object",
        "loads": "object",
        "registryReqconsEnabled": true
    },
    "relationships": {
        "organisation": [{"id": "string", "type": "string"}],
        "distributor": [{"id": "string", "type": "string"}],
        "meterOwner": [{"id": "string", "type": "string"}],
        "gxp": [{"id": "string", "type": "string"}],
        "retailer": [{"id": "string", "type": "string"}],
        "verificationMethod": [{"id": "string", "type": "string"}],
        "programmes": [{"id": "string", "type": "string"}],
        "registrations": [{"id": "string", "type": "string"}],
        "availabilities": [{"id": "string", "type": "string"}]



}
```

Attributes

The following attributes can be submitted when creating a new site

| Name | Required | Type | Description | Example |
|---|---|---|---|---|
| name | true | string | Unique name for the site | ACME Sawmill |
| icpNumber | true | string | ICP Number | 8671784589NI73E |
| address | true | string | Street address of site | 1 Main street |
| status | true | string | Site active status | One of:<br>● Active<br>● Inactive |
| flowDirection | true | string | Flow direction to observe for verification | One of:<br>● I<br>● X<br>● X-I |
| consumerAuthorizationCode | false | string | Consumer Authorisation Code for EIEP Hub | ABC123 |
| consumerNo | false | string | Consumer Number Code for EIEP Hub | 100435833 |
| customerName | false | string | Customer name for EIEP Hub | John Smith |
| registryReqconsEnabled | false | boolean | Determines if load data should be automatically retrieved from EIEP hub | true/false |
| loads | true | object | The breakdown of site loads (measured in kW) by load type. Keys in the object must be valid load types | {"Lighting" => 100, "Refrigeration" => 200} |

The following relationships can be submitted for a site

| Name | Required | Type | Description | Example |
|---|---|---|---|---|
| retailer | true | object | Retailer for the site | {"data": {"id": 1, "type": "organisations"} } |
| distributor | true | object | Distributor for the site | {"data": {"id": 1, "type": "organisations"} } |
| meterOwner | true | object | Meter Owner for the site | {"data": {"id": 1, "type": "organisations"} } |
| gxp | true | object | GXP for the site | {"data": {"id": 1, "type": "gxps"}} |
| verificationMethod | true | object | Verification Method for the site | {"data": {"id": 1, "type": "verificationMethods"}} |

ℹ **Where can I find IDs for these entities?** Use the GET /organisations/{type}, GET /gxps, GET /load_types and GET /verification_methods endpoints to identify the needed related entities.

### Filters

The following filters can be applied on the index endpoint (GET /sites)

| Name | Description | Example |
|---|---|---|
| name | Filter by name (accepts partial match) | ACME |
| active | Filter by active status | true/false |

### Fields

The following fields (and any subset thereof) can be requested for a site

```
"name","icpNumber","meterId","address",
"status","kwAmount","flowDirection","tags",
"loads","consumerAuthorisationCode","consumerNo","customerName",
"organisation","gxp","retailer","distributor",
"meterOwner","verificationMethod","programmes","registrations",
"availabilities","registryReqconsEnabled"
```

### Relationships

The following relationships & corresponding fields can be optionally included when requesting sites.

```
"programmes": ["name","organisation"],
"registrations": ["status"],
"organisations": ["name"],
"verificationMethods": ["name"],
"gxps": ["code","tags"],
"availabilities": ["from","to","weekDays"]
```

### Ordering

Not supported

## Endpoints

### GET /sites

Get a list of all sites accessible to the current user

### Permissions

Requires the **read_sites** permission for the parent organisation

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/sites</td><td><b>Include, filter[], fields[], page[number], page[size]</b></td><td>GET</td><td>required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2">None</td><td colspan="2">{<br>  "data": [<br>    {<br>      "id": "1",<br>      "type": "sites",</td></tr>
</table>

| | |
|---|---|
| | "attributes": {<br>  "name": "ST-PIO-1",<br>  "icpNumber": "0007110201RN312",<br>   ...<br>},<br>"relationships": {...}<br>},<br>{<br>  "id": "2",<br>  "type": "sites",<br>  "attributes": {<br>    "name": "ST-PIO-2",<br>    "icpNumber": "0007100766RNCCC",<br>    "meterId": "215504861",<br>     …<br>  },<br>  "relationships": {...}<br>}<br>]<br>} |

### GET /sites/{id}

Get a single site

#### Permissions

Requires the **write_sites** permission for the parent organisation

| **Endpoint** | **Query Options** | **Method** | **Authentication** |
|---|---|---|---|
| /sites/{id} | **Include, filter, fields** | GET | Required |
| **Payload** | | **Response** | |
| None | | {<br>  "data": {<br>    "id": "1",<br>    "type": "sites",<br>    "attributes": {<br>      "name": "ST-PIO-1",<br>      "icpNumber": "0007110201RN312"<br>       ....<br>    },<br>    "relationships": {...}<br>  }<br>} | |

### **POST** /organisations/{organisation_id}/sites/{id}

Create a site

ℹ **Where can I find my organisation ID?** Look at the Tokens endpoint for more info on how to decode your token and extract your organisation ID

Permissions

Requires the **write_sites** permission for the parent organisation

| **Endpoint** | **Query Options** | **Method** | **Authentication** |
|---|---|---|---|
| /organisations/{organisation_id}/sites/{id} | **Include, filter[], fields[]** | POST | Required |

**Payload**

```
{
  "data": {
    "type": "sites",
    "attributes": {
      "status": "active",
      "type": "sites",
      "name": "ACME Sawmill",
      "icpNumber": "8671784589NI73E",
      "meterId": "1007243",
      "consumerAuthorisationCode": null,
      "consumerNo": null,
      "customerName": null,
      "address": "1 Main Street",
      "flowDirection": "X-I",
      "loads": {
        "Lighting": 100,
        "Refrigeration": 100
      },
      "registryReqconsEnabled": false
    },
    "relationships": {
      "gxp": {
        "data": {"id": "19","type": "gxps"}
      },
      "retailer": {
        "data": {"id": "42","type":
"organisations"}
      },
      "distributor": {
        "data": {"id": "62","type":
"organisations"}
      },
      "meterOwner": {
        "data": {"id": "6","type":
"organisations"}
      },
      "verificationMethod": {
        "data": {"id":
"1","type":"verificationMethods"}
      }
    }
  }
}
```

**Response**

```
{
  "data": {
    "id": "1",
    "type": "sites",
    "attributes": {
      "name": "ACME Sawmill",
      "icpNumber": "8671784589NI73E"
       ....
    },
    "relationships": {...}
  }
}
```

### PUT /sites/{id}

Update a site

### Permissions

Requires the **write_sites** permission for the parent organisation

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /sites/{id} | **Include, filter, fields** | PUT | Required |

**Payload**

```
{
  "data": {
    "type": "sites",
    "attributes": {
      "organisationId": "1",
      "status": "active",
      "type": "sites",
      "name": "ACME Sawmill",
      "icpNumber": "8671784589NI73E",
      "meterId": "1007243",
      "consumerAuthorisationCode": null,
      "consumerNo": null,
      "customerName": null,
      "address": "1 Main Street",
      "flowDirection": "X-I",
      "loads": {
        "Lighting": 100,
        "Refrigeration": 100
      },
      "registryReqconsEnabled": false
    },
    "relationships": {
      "gxp": {
        "data": {"id": "19","type": "gxps"}
      },
      "retailer": {
        "data": {"id": "42","type":
"organisations"}
      },
      "distributor": {
        "data": {"id": "62","type":
"organisations"}
      },
      "meterOwner": {
        "data": {"id": "6","type":
"organisations"}
      },
      "verificationMethod": {
        "data": {"id":
"1","type":"verificationMethods"}
      }
    }
  }
}
```

**Response**

```
{
  "data": {
    "id": "1",
    "type": "sites",
    "attributes": {
      "name": "ACME Sawmill",
      "icpNumber": "8671784589NI73E"
       ….
    },
    "relationships": {...}
  }
}
```

### DELETE /sites/{id}

Delete a site

#### Permissions

Requires the **write_sites** permission for the parent organisation

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/sites/{id}</td><td><b>Include, filter, fields</b></td><td>DELETE</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2"><pre>None</pre></td><td colspan="2"><pre>{
  "data": {
    "id": "1",
    "type": "sites",
    "attributes": {
      "name": "ACME Sawmill",
      "icpNumber": "8671784589NI73E"
       ….
    },
    "relationships": {...}
  }
}</pre></td></tr>
</table>

## Registrations

The **registration** endpoints allow you to manage participant registrations in the DERMS. A registration **must** be admin-approved before it becomes effective.

Any registrations created by a participant will exist in a draft state. A registration can be submitted for approval to an administrator.

### Resource

#### Schema

```
{
  "data": {
    "id": "string",
    "type": "string",
    "attributes": {
      "name": "string",
      "minimumLeadTime": "string",
      "tags": "string",
      "startDate": "2001-01-01",
      "endDate": "2001-01-01",
      "fixedPrice": 0,
      "availabilityFee": 0,
      "prepurchasedHours": 0,
      "indicativePrice": 0,
      "status": "string",
      "kwAmount": "string",
      "availableDR": "string",
      "weeklyAvailabilityBitmask": "string",
      "useAggregateCbl": true,
      "readOnly": "string",
      "rejectionReason": "string"
    },
    "relationships": {
      "organisation": [{"id": "string","type": "string"}],
      "programme": [{"id": "string","type": "string"}],
      "substation": [{"id": "string","type": "string"}],
      "verificationMethod": [{"id": "string","type": "string"}],
      "initialEstablishmentFee": [{"id": "string","type": "string"}],
      "finalEstablishmentFee": [{"id": "string","type": "string"}],
      "sites": [{"id": "string","type": "string"}],
      "targetedDevices": [{"id": "string","type": "string"}]
    }
  }
}
```

Attributes

The following attributes can be submitted when creating a new registration

| Name | Required | Type | Description | Example |
|---|---|---|---|---|
| name | true | string | Unique name for the registration | ACME Registration 01 |
| startDate | true | string | The start date of the registrations (iso8601 format) | 2001-01-01 |
| endDate | true | string | The end date of the registration (iso8601 format) | 2002-01-01 |
| fixedPrice | false (unless required by programme) | number | The fixed price to be offered for this registration to participate in events (measured in $/MWh) | One of:<br>● Active<br>● Inactive |
| availabilityFee | false (unless required by programme) | number | The availability fee for the registration (measured in $) | 2000 |
| prepurchasedHours | false (unless required by programme) | number | The number of prepurchased hours | 80 |
| indicativePrice | true (only if price responsive registration) | number | The indicative price to be offered for this registration (measured in $/MWh) | 1000 |
| useAggregateCbl | false | boolean | Customer name for EIEP Hub | John Smith |
| initialEstablishmentFee | false (unless required by programme) | object | The date and amount of the initial establishment fee for this registration | {"amount":7.5,"date":"2001-01-01"} |
| finalEstablishmentFee | false (unless required by programme) | object | The date and amount of the final establishment fee for this registration | {"amount":7.5,"date":"2002-01-01"} |

### Relationships

The following relationships can be submitted for a registration

| Name | Required | Type | Description | Example |
|---|---|---|---|---|
| program | true | object | The programme under which to register | {"data": {"id": 1, "type": "programmes"}} |
| sites | true (unless substation registration) | array of object | The sites to include in this registration | [{"data": {"id": 1, "type": "organisations"}}] |
| substation | true (unless sites registration) | object | The substation to register | {"data": {"id": 1, "type": "substations"}} |
| verificationMethod | false (unless **useAggregateCbl** is true) | object | Verification Method for the registration | {"data": {"id": 1, "type": "verificationMethods"}} |

### Filters

The following filters can be applied on the index endpoint (GET /registrations)

| Name | Description | Example |
|---|---|---|
| name | Filter by name (accepts partial match) | ACME |
| status | Filter by status | one of: draft, submitted, active, inactive |
| programmeName | Filter by programme name (accepts partial match) | Price Responsive Programme |
| GXP | Filter by GXP | ABY0111 |
| priceResponsive | Include only price responsive registrations | true |
| nonPriceResponsive | Include only price non-price responsive registrations | true |
| from | Include only registrations active from a given date | 2001-01-01 |
| to | Include only registrations active before a given date | 2001-01-01 |

### Fields

The following fields (and any subset thereof) can be requested for a registration

```
"name","minimumLeadTime","startDate","endDate","fixedPrice","availabilityFe
e","prepurchasedHours","indicativePrice","status","rejectionReason","readOn
ly","sites","substation","organisation","programme","tags","gxp","distribut
or","kwAmount","availableDR","signalCapacities","targetedDevices","weeklyAv
ailabilityBitmask","useAggregateCbl","verificationMethod","initialEstablish
mentFee","finalEstablishmentFee"
```

### Relationships

The following relationships & corresponding fields can be optionally included when requesting registrations.

```
"organisation": ["name"]
"programme": ["name"]
"sites": ["name", "kwAmount","gxp", "distributor"]
"sites.gxp": ["name", "code", "tags"]
"sites.distributor": ["name"]
"substation": ["name", "kwAmount","gxp", "distributor"]
"substation.gxp": ["name", "code", "tags"]
"initialEstablishmentFee": ["date", "amount", "bcti"]
"finalEstablishmentFee": ["date", "amount", "bcti"]
"initialEstablishmentFee.bcti": ["name"]
"finalEstablishmentFee.bcti": ["name"]
"verificationMethod": ["name"]
```

### Ordering

Not supported

## Endpoints

### GET/registrations

Get a list of all registrations accessible to the current user

### Permissions

Requires the **read_registration** permission for the parent organisation

| **Endpoint** | **Query Options** | **Method** | **Authentication** |
|---|---|---|---|
| /registrati | **Include, filter,** | GET | required |

<table>
<tr><td>ons</td><td><b>fields,<br>page[number],<br>page[size]</b></td><td></td><td></td></tr>
<tr><td colspan="2"><b>Payload</b></td><td colspan="2"><b>Response</b></td></tr>
<tr><td colspan="2"><pre>None</pre></td><td colspan="2"><pre>{
  "data": [
    {
      "id": "1",
      "type": "registrations",
      "attributes": {
        "name": "RG-PIO-1",
        "startDate": "2019-01-01",
          ...
      },
      "relationships": {...}
    },
    {
      "id": "2",
      "type": "registrations",
      "attributes": {
        "name": "RG-PIO-2",
        "startDate": "2019-02-01",
         …
      },
      "relationships": {...}
    }
 ]
}</pre></td></tr>
</table>

### GET /registrations/{id}

Get a single registration

#### Permissions

Requires the **read_registration** permission for the parent organisation

<table>
<tr><td><b>Endpoint</b></td><td><b>Query Options</b></td><td><b>Method</b></td><td><b>Authentication</b></td></tr>
<tr><td>/registration/{id}</td><td><b>Include, filter,<br>fields</b></td><td>GET</td><td>Required</td></tr>
<tr><td colspan="2"><b>Payload</b></td><td colspan="2"><b>Response</b></td></tr>
<tr><td colspan="2"><pre>None</pre></td><td colspan="2"><pre>{
  "data": {
    "id": "1",
    "type": "registrations",
    "attributes": {
      "name": "RG-PIO-1",
      "startDate": "2019-01-01",
       ....
    },
    "relationships": {...}
  }
}</pre></td></tr>
</table>

### **POST** /organisations/{organisation_id}/registrations/{id}

Create a registration. Note that a new registration begins with a **draft** status. To submit the registration for review look at the events endpoint.

#### Permissions

Requires either the **write_programmes** permission for the parent organisation or the **write_registration** permission for the organisation or registration in question

> ℹ **Where can I find my organisation ID?** Look at the Tokens endpoint for more info on how to decode your token and extract your organisation ID

| **Endpoint** | **Query Options** | **Method** | **Authentication** |
|---|---|---|---|
| /organisations/{organisation_id}/registrations/{id} | **Include, filter, fields** | POST | Required |

**Payload**

```
{
  "data": {
    "attributes": {
      "name": "ACME Reg",
      "startDate": "2019-07-02",
      "endDate": "2020-07-02",
      "organisationId": "1",
      "initialEstablishmentFee": {
        "amount": 7.5,
        "date": "2019-07-02T20:49:32.821Z"
      },
      "finalEstablishmentFee": {
        "amount": 7.5,
        "date": "2020-07-02"
      },
      "indicativePrice": 50,
      "useAggregateCbl": true,
      "fixedPrice": 15,
      "availabilityFee": 15,
      "prepurchasedHours": 15
    },
    "relationships": {
      "sites": {"data": [{"id": "9","type":
"sites"}]},
      "programme": {"data": {"id": "2","type":
"programmes"}},
      "verificationMethod": {"data": {"id":
"3","type": "verificationMethods"}}
    }
  }
}
```

**Response**

```
{
  "data": {
    "id": "1",
    "type": "registrations",
    "attributes": {
      "name": "ACME Reg",
      "startDate": "2019-07-02",
      "endDate": "2020-07-02",
      "status": "draft",
       ....
    },
    "relationships": {...}
  }
}
```

### POST /registrations/{id}/events

Trigger an event on a registration. The only event that can be triggered by a participant is the **submit** event

#### Permissions

Requires either the **write_programmes** permission for the parent organisation or the **write_registration** permission for the organisation or registration in question

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/registrations/{id}/events</td><td><b>None</b></td><td>POST</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2"><pre>
{
  "data": {
    "attributes": {
      "name": "submit"
    }
  }
}
</pre></td><td colspan="2"><pre>
{
  "data": {
    "id": "",
    "type": "events",
    "attributes": {
      "name": "submit",
      "options": {}
    },
    "relationships": {
      "target": {
        "meta": {
          "included": false
        }
      }
    }
  },
  "jsonapi": {
    "version": "1.0"
  }
}
</pre></td></tr>
</table>

### PUT /registrations/{id}

Update a registration. A participant can only update a registration while it is in draft status

#### Permissions

Requires either the **write_programmes** permission for the parent organisation or the **write_registration** permission for the organisation or registration in question

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/sites/{id}</td><td><b>Include, filter, fields</b></td><td>PUT</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2"><pre>
{
  "data": {
    "attributes": {
</pre></td><td colspan="2"><pre>
{
  "data": {
    "id": "1",
</pre></td></tr>
</table>

<table>
<tr>
<td><pre>      "name": "ACME Reg (v2)",
      "startDate": "2019-07-02",
      "endDate": "2021-07-02",
    }
  }
}</pre></td>
<td><pre>    "type": "registrations",
    "attributes": {
      "name": "ACME Reg (v2)",
      "startDate": "2019-07-02",
      "endDate": "2021-07-02",
      "status": "draft",
       ….
    },
    "relationships": {...}
  }
}</pre></td>
</tr>
</table>

### DELETE /registrations/{id}

Delete a registration

#### Permissions

Requires either the **write_programmes** permission for the parent organisation or the **write_registration** permission for the organisation or registration in question

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/sites/{id}</td><td><b>Include, filter, fields</b></td><td>DELETE</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr>
<td colspan="2"><pre>None</pre></td>
<td colspan="2"><pre>{
  "data": {
    "id": "1",
    "type": "registrations",
    "attributes": {
      "name": "ACME Reg (v2)",
      "startDate": "2019-07-02",
      "endDate": "2021-07-02",
      "status": "draft",
       ….
    },
    "relationships": {...}
  }
}</pre></td>
</tr>
</table>

# Events

## Resource

### Schema

```
{
  "data": {
    "id": "string",
    "type": "string",
    "attributes": {
      "targetKw": 0,
      "mwhPrice": 0,
      "accepted": true,
      "minimumLeadTime": "string",
      "optedIn": "string",
      "status": "string",
      "offerKw": "string",
      "offeredAutoDRKw": "string",
      "offerPrice": "string",
      "drLevel": "string"
    },
    "relationships": {
      "parentEvent": [{"id": "string","type": "string"}],
      "registration": [{"id": "string","type": "string"}],
      "lastPayment": [{"id": "string","type": "string"}],
      "bcti": [{"id": "string","type": "string"}],
      "offer": [{"id": "string","type": "string"}],
      "programme": [{"id": "string","type": "string"}]
    }
  }
}
```

### Filters

The following filters can be applied on the index endpoint (GET /child_events)

| Name | Description | Example |
|---|---|---|
| name | Filter by name (accepts partial match) | ACME |
| status | Filter by status | one of:<br>● completed<br>● active<br>● meter_data_required<br>● meter_data_submitte |

| | | |
|---|---|---|
| | | d<br>● offer_window_closed<br>● offer_window_open<br>● payment_complete<br>● payment_pending<br>● declined<br>● scheduled<br>● withdrawn<br>● ineligible |
| programme | Filter by programme name (accepts partial match) | Price Responsive Programme |
| registration | Filter by registration name (accepts partial match) | |
| priceResponsive | Include only price responsive events | true |
| nonPriceResponsive | Include only price non-price responsive child events | true |
| from | Include only events active from a given date and time | 2001-01-01T12:00 |
| to | Include only events active before a given date and time | 2001-01-01T14:00 |

### Fields

The following fields (and any subset thereof) can be requested for a child event

```
"targetKw","mwhPrice","parentEvent","minimumLeadTime","registration","offer
","programme","optedIn","status","lastPayment","offerKw","offeredAutoDRKw",
"offerPrice","drLevel"
```

### Relationships

The following relationships & corresponding fields can be optionally included when requesting child events.

```
"parent_events":[
  "start","end","offerWindowStart",
   "offerWindowEnd","status","organisation","name"
],
"programmes":["name","type","organisation"],
"organisations":["name"],
"registrations":[
  "organisation","name","sites","substation",
  "minimumLeadTime","indicativePrice","fixedPrice",
  "availabilityFee","prepurchasedHours","availableDR",
  "targetedDevices","weeklyAvailabilityBitmask"
],
"offers":[
  "offerKw","nonAutoDRKw","offerPrice","status",
  "createdAt","updatedAt","drLevel","autoDREvents"
],
"substations":["name","gxp"],
"sites":["name","gxp"],
"payments":[
  "performancePercentage","bcti","settlementAmount","createdAt"
],
"bctis":["createdAt","filename","invoiceUrlPath"],
"gxps":["name","code","tags"],
"auto_dr_events":["estimatedKwAmount","device"]
```

#### Ordering

Not supported

### Endpoints

#### GET /child_events

Get a list of all child_events. Remember to use the **include** parameter to include the associated registration, parent event and offer if required

#### Permissions

Requires the **read_registration** permission for the target registration

| **Endpoint** | **Query Options** | **Method** | **Authentication** |
|---|---|---|---|
| /child_events | **Include, filter, fields, page[size], page[number]** | GET | Required |

| **Payload** | **Response** |
|---|---|
| `None` | `{`<br>`  "data": [`<br>`    {`<br>`      "id": "997",`<br>`      "type": "childEvents",`<br>`      "attributes": {` |

<table>
<tr><td></td><td><pre>
        "targetKw": 280,
        "mwhPrice": 1000,
        "minimumLeadTime": 60,
        "optedIn": true,
        "status": "meter_data_required",
        "offerKw": 280,
        "offeredAutoDRKw": 0,
        "offerPrice": 1000,
        "drLevel": null
      },
      "relationships": {...}
    },
    {
      "id": "996",
      "type": "childEvents",
      "attributes": {
        "targetKw": 280,
        "mwhPrice": 1000,
        "minimumLeadTime": 60,
        "optedIn": true,
        "status": "meter_data_required",
        "offerKw": 280,
        "offeredAutoDRKw": 0,
        "offerPrice": 1000,
        "drLevel": null
      },
      "relationships": {...}
    }
  ],
  "meta": {
    "totalPages": 1,
    "count": 2,
    "page": 1
  },
  "jsonapi": {
    "version": "1.0"
  }
}
</pre></td></tr>
</table>

### GET /child_events/{id}

Get a single child_event. Remember to use the **include** parameter to include the associated registration, parent event and offer if required

#### Permissions

Requires either the **read_registration** permission for the target registration

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/child_events/{id}</td><td><b>None</b></td><td>GET</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2"><pre>None</pre></td><td colspan="2"><pre>
{
  "data": {
    "id": "997",
</pre></td></tr>
</table>

```
    "type": "childEvents",
    "attributes": {
      "targetKw": 280,
      "mwhPrice": 1000,
      "minimumLeadTime": 60,
      "optedIn": true,
      "status": "meter_data_required",
      "offerKw": 280,
      "offeredAutoDRKw": 0,
      "offerPrice": 1000,
      "drLevel": null
    },
    "relationships": {...}
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

## Offers

### Resource

#### Schema

```
{
  "data": {
    "id": "string",
    "type": "string",
    "attributes": {
      "status": "string",
      "offerKw": 0,
      "nonAutoDRKw": 0,
      "autoDRKw": "string",
      "createdAt": "2001-01-01 00:30+12",
      "updatedAt": "2001-01-01 00:30+12",
      "offerPrice": 0,
      "targetKw": "string",
      "targetPrice": "string",
      "drLevel": "string"
    },
    "relationships": {
      "childEvent": [{"id": "string","type": "string"}],
      "autoDREvents": [{"id": "string","type": "string"}]
    }
  }
}
```

#### Filters

Not supported

#### Fields

The following fields (and any subset thereof) can be requested for an offer

```
"status","offerKw","nonAutoDRKw","autoDRKw","offerPrice","drLevel","childEvent","targetKw","targetPrice","autoDREvents"
```

#### Relations

Not supported

### Ordering

Not supported

## Endpoints

### GET /offers

Get a list of all offers. Offers can be requested independently but make more sense when requested along-side child events.

### Permissions

Requires the **read_registration** permission for the target registration

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/offers</td><td><b>page[size],<br>page[number]</b></td><td>GET</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2"><pre>None</pre></td><td colspan="2"><pre>
{
  "data": [
    {
      "id": "884",
      "type": "offers",
      "attributes": {
        "status": "opted_in",
        "offerKw": 280,
        "nonAutoDRKw": 0,
        "autoDRKw": 0,
        "offerPrice": 1000,
        "targetKw": 280,
        "targetPrice": 1000,
        "drLevel": null
      },
      "relationships": {
        "childEvent": {"meta": {"included": false}},
        "autoDREvents": {"meta": {"included": false}}
      }
    }
  ],
  "meta": {
    "totalPages": 1,
    "count": 1,
    "page": 1
  },
  "jsonapi": {
    "version": "1.0"
  }
}
</pre></td></tr>
</table>

### GET /offers/{id}

Get a single child_event. Offers can be requested independently but make more sense when requested along-side child events.

#### Permissions

Requires the **read_registration** permission for the target registration

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/offers/{id}</td><td><b>None</b></td><td>GET</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2"><pre>None</pre></td><td colspan="2"><pre>{
  "data": {
    "id": "884",
    "type": "offers",
    "attributes": {
      "status": "opted_in",
      "offerKw": 280,
      "nonAutoDRKw": 0,
      "autoDRKw": 0,
      "offerPrice": 1000,
      "targetKw": 280,
      "targetPrice": 1000,
      "drLevel": null
    },
    "relationships": {
      "childEvent": {"meta":
      "autoDREvents": {"meta": {"included": false}}
    }
  },
  "meta": {
    "totalPages": 1,
    "count": 1,
    "page": 1
  },
  "jsonapi": {
    "version": "1.0"
  }
}</pre></td></tr>
</table>

### PUT /offers/{id}

Use the PUT /offers/{id} endpoint to respond to an offer by either opting in (optionally amending the offer) or opting out

#### Permissions

Requires either the **write_registration** permission for the target registration

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|

| /offers/{id} | **None** | PUT | Required |
|---|---|---|---|

**Payload**

```
{
  "data": {
    "attributes": {
      "status": "opted_in",
      "offerKw": 900,
      "offerPrice": 200
    }
  }
}
```

**Response**

```
{
  "data": {
    "id": "884",
    "type": "offers",
    "attributes": {
      "status": "opted_in",
      "offerKw": 200,
      "nonAutoDRKw": 0,
      "autoDRKw": 0,
      "offerPrice": 200,
      "targetKw": 280,
      "targetPrice": 1000,
      "drLevel": null
    },
    "relationships": {
      "childEvent": {"meta":
      "autoDREvents": {"meta": {"included": false}}
    }
  },
  "meta": {
    "totalPages": 1,
    "count": 1,
    "page": 1
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

## Tasks

A user can query the system for any tasks that are awaiting their input.
The only task that may require the input of a participant is the requirement to submit meter data for sites that have participated in an event.

### Requests

GET /tasks Get a list of all tasks

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/tasks</td><td></td><td>GET</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2"><pre>None</pre></td><td colspan="2"><pre>{
  "data": {
    "id": "1",
    "type": "userTaskLists",
    "attributes": {
      "childEventsPendingMeterData": 5
    }
  ],
  "jsonapi": {
    "version": "1.0"
  }
}</pre></td></tr>
</table>

*To follow up on these tasks the client can submit a second API request to the GET /child_events endpoint to discover* ***which*** *child events require meter data*

## Substations

The **substation** endpoints allow you to manage participant sites in the DERMS.

### Resource

#### Schema

```
{
  "data": {
      "id": "string",
      "type": "string",
      "attributes": {
        "name": "string",
        "status": "string",
        "address": "string",
        "flowDirection": "string",
        "kwAmount": 0,
         "tags": "object",
        "loads": "object"
      },
      "relationships": {
        "organisation": [{"id": "string", "type": "string"}],
        "distributor": [{"id": "string", "type": "string"}],
        "meterOwner": [{"id": "string", "type": "string"}],
        "gxp": [{"id": "string", "type": "string"}],
        "verificationMethod": [{"id": "string", "type": "string"}],
        "programmes": [{"id": "string", "type": "string"}],
        "registrations": [{"id": "string", "type": "string"}]
     }
  }
}
```

Attributes

The following attributes can be submitted when creating a new substation

| Name | Required | Type | Description | Example |
|---|---|---|---|---|
| name | true | string | Unique name for the substation | ACME Sawmill |
| address | true | string | Street address of substation | 1 Main street |
| status | true | string | Substation active status | One of:<br>● Active<br>● Inactive |
| flowDirection | true | string | Flow direction to observe for verification | One of:<br>● I<br>● X<br>● X-I |
| loads | true | object | The breakdown of substation loads (measured in kW) by load type. Keys in the object must be valid load types | {"Lighting" => 100, "Refrigeration" => 200} |

The following relationships can be submitted for a substation

| Name | Required | Type | Description | Example |
|---|---|---|---|---|
| distributor | true | object | Distributor for the substation | {"data": {"id": 1, "type": "organisations"} } |
| meterOwner | true | object | Meter Owner for the substation | {"data": {"id": 1, "type": "organisations"} } |
| gxp | true | object | GXP for the substation | {"data": {"id": 1, "type": "gxps"}} |
| verificationMeth | true | object | Verification | {"data": {"id": 1, |

| | | | | |
|---|---|---|---|---|
| od | | | Method for the substation | "type": "verificationMethods"}} |

ℹ **Where can I find IDs for these entities?** Use the GET /organisations/{type}, GET /gxps, GET /load_types and GET /verification_methods endpoints to identify the needed related entities.

### Filters

The following filters can be applied on the index endpoint (GET /substations)

| Name | Description | Example |
|---|---|---|
| name | Filter by name (accepts partial match) | ACME |
| active | Filter by active status | true/false |

### Fields

The following fields (and any subset thereof) can be requested for a substation

```
"name","address", "status","kwAmount","flowDirection","tags",
"loads","organisation","gxp","distributor",
"meterOwner","verificationMethod","programmes","registrations"
```

### Relationships

The following relationships & corresponding fields can be optionally included when requesting substations.

```
"programmes": ["name","organisation"],
"registrations": ["status"],
"organisations": ["name"],
"verificationMethods": ["name"],
"gxps": ["code","tags"],
```

### Ordering

Not supported

## Endpoints

### GET /substations

Get a list of all substations accessible to the current user

#### Permissions

Requires the **read_substations** permission for the parent organisation

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /substations | **Include, filter, fields, page[number], page[size]** | GET | required |

| Payload | Response |
|---|---|
| None | { <br> "data": [ <br> { <br> "id": "1", <br> "type": "substations", <br> "attributes": { <br> "name": "SBST-PIO-1", <br> "address": "1 Main Street", <br> ... <br> }, <br> "relationships": {...} <br> }, <br> { <br> "id": "2", <br> "type": "substations", <br> "attributes": { <br> "name": "SBST-PIO-2", <br> "address": "1 Key Road", <br> … <br> }, <br> "relationships": {...} <br> } <br> ] <br> } |

### GET /substations/{id}

Get a single substation

#### Permissions

Requires either the **read_substations** permission for the parent organisation

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /substations/{id} | **Include, filter, fields** | GET | Required |

| Payload | Response |
|---|---|
| None | {<br>  "data": {<br>    "id": "1",<br>    "type": "substations",<br>    "attributes": {<br>      "name": "SBST-PIO-1",<br>      "address": "1 Main Street",<br>       ….<br>    },<br>    "relationships": {...}<br>  }<br>} |

**POST** /organisations/{organisation_id}/substations/{id}

Create a substation

Permissions

Requires the **write_substations** permission for the parent organisation

ℹ **Where can I find my organisation ID?** Look at the Tokens endpoint for more info on how to decode your token and extract your organisation ID

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /organisations/{organisation_id}/substations/{id} | **Include, filter, fields** | POST | Required |
| **Payload** | **Response** | | |
| {<br>  "data": {<br>    "type": " ",<br>    "attributes": {<br>      "status": "active",<br>      "type": "substations",<br>      "name": "ACME Sub",<br>      "address": "1 Main Street",<br>      "flowDirection": "X-I",<br>      "loads": {<br>        "Lighting": 100,<br>        "Refrigeration": 100<br>      }<br>    },<br>    "relationships": {<br>      "gxp": {<br>        "data": {"id": "19","type": "gxps"}<br>      }, | {<br>  "data": {<br>    "id": "1",<br>    "type": "substations",<br>    "attributes": {<br>      "name": ACME Sub",<br>      "address": "1 Main Street",<br>       ….<br>    },<br>    "relationships": {...}<br>  }<br>} | | |

```
      "distributor": {
        "data": {"id": "62","type":
"organisations"}
      },
      "meterOwner": {
        "data": {"id": "6","type":
"organisations"}
      },
      "verificationMethod": {
        "data": {"id":
"1","type":"verificationMethods"}
      }
    }
  }
}
```

### PUT /substations/{id}

Update a substation

#### Permissions

Requires the **write_substations** permission for the parent organisation

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /substations/{id} | **Include, filter, fields** | PUT | Required |

**Payload**

```
{
  "data": {
    "type": "substations",
    "attributes": {
      "status": "active",
      "type": "sites",
      "name": "ACME Sub (v2)"
    }
  }
}
```

**Response**

```
{
  "data": {
    "id": "1",
    "type": "substations",
    "attributes": {
      "name": "ACME Sub (v2)",
       ….
    },
    "relationships": {...}
  }
}
```

### DELETE /substations/{id}

Delete a substation

#### Permissions

Requires the **write_substations** permission for the parent organisation

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /substations/{id} | **Include, filter, fields** | DELETE | Required |

| Payload | Response |
|---|---|
| None | {<br>  "data": {<br>    "id": "1",<br>    "type": "substations",<br>    "attributes": {<br>      "name": "ACME Sub"<br>       ….<br>    },<br>    "relationships": {...}<br>  }<br>} |

## Devices

### Resource

#### Schema

```
{
  "data": {
    "id": "string",
    "type": "string",
    "attributes": {
      "name": "string",
      "venId": "string",
      "type": "string",
      "make": "string",
      "model": "string",
      "serialNumber": "string",
      "status": "string",
      "tags": "string",
      "lastHeartbeatAt": "string",
      "online": true,
      "signalCapacities": "string"
    },
    "relationships": {
      "site": [{"id": "string","type": "string"}],
      "deviceSignals": [{"id": "string","type": "string"}]
    }
  }
}
```

Attributes

The following attributes can be submitted when creating a new device

| Name | Required | Type | Description | Example |
|---|---|---|---|---|
| name | true | string | Unique name for the substation | ACME Sawmill |
| ven_id, ambi_id or dreds_device_id | true | string | The unique device identifier (conditional on the device type) | 43587123 |
| type | true | string | The device type | One of:<br>● Ambi::Device<br>● FuturePoint::Device<br>● Oadr::Device |
| dreds_class | Only if dreds device | string | Flow direction to observe for verification | One of:<br>● air_conditioner<br>● strip_heaters<br>● water_heaters<br>● pool_pumps<br>● smart_appliances<br>● irrigation_pumps<br>● managed_commercial_industrial<br>● simple_misc<br>● exterior_lighting<br>● electric_vehicle<br>● generation_systems |
| make | false | string | The device make | ACME |
| model | false | string | The device model | ACM014 |
| serial number | false | string | The device serial number | LKJ345982 |
| status | true | string | Device status | active or inactive |
| signalCapacities | true | Object | A key value pair describing the capacity of the device at each supported DR signal | `{`<br>`  "1": "2.0",`<br>`  "2": "1.0",`<br>`  "3": "1.0"`<br>`}` |

### Filters

The following filters can be applied on the index endpoint (GET /devices)

| Name | Description | Example |
|---|---|---|
| name | Filter by name (accepts partial match) | ACME |
| active | Filter by active status | true/false |

### Fields

The following fields (and any subset thereof) can be requested for a device

```
"name","venId","ambiId", "dredsDeviceId","type","dreds_class","make",
"model","serialNumber","status","site","lastHearbeatAt","online","signalCap
acities","deviceSignals","canSendTestEvent","testEventParams"
```

### Relations

The following relationships & corresponding fields can be optionally included when requesting substations.

```
  "sites": [
      "name",
      "organisation"
  ],
  "organisations": [
      "name"
  ],
  "oadr_registrations": [
      "venName",
      "profileName",
      "transportName",
      "transportAddress",
      "reportOnly",
      "xmlSignature",
      "httpPullModel"
  ]
```

### Ordering

Not supported

## Endpoints

### GET /sites/{site_id}/devices

Get a list of all devices for a site

### Permissions

Requires the **read_devices** permission for the parent organisation

| **Endpoint** | **Query Options** | **Method** | **Authentication** |
|---|---|---|---|
| /sites/{site_id}/devices | **Include, filter, fields, page[size], page[number]** | GET | Required |

| **Payload** | **Response** |
|---|---|
| None | {<br>  "data": [{<br>    "id": "2",<br>    "type": "ambiDevices",<br>    "attributes": {<br>      "name": "Test Device",<br>      "ambiId": "3290458203947453",<br>      "type": "Ambi::Device",<br>      "make": null,<br>      "model": null,<br>      "serialNumber": null,<br>      "status": "active",<br>      "lastHeartbeatAt": "2019-07-04T14:01:13.786+12:00",<br>      "online": true,<br>      "signalCapacities": {<br>        "1": "2.0",<br>        "2": "1.0",<br>        "3": "1.0"<br>      },<br>      "canSendTestEvent": true,<br>      "testEventParams": {<br>        "minLead": 2100,<br>        "minLength": 900,<br>        "bufferTime": 1,<br>        "signals": [<br>          {<br>            "id": 5,<br>            "value": "1"<br>          },<br>          {<br>            "id": 6,<br>            "value": "2"<br>          },<br>          {<br>            "id": 7,<br>            "value": "3"<br>          } |

<table>
<tr><td></td><td><pre>
        ]
      }
    },
    "relationships": {...}
  }],
  "jsonapi": {
    "version": "1.0"
  }
}
</pre></td></tr>
</table>

### POST /sites/{site_id}/devices

Create a device underneath a site

### Permissions

Requires the **write_devices** permission for the parent organisation

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/sites/{site_id}/devices</td><td><b>Include, filter, fields</b></td><td>POST</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2"><pre>
{
  "data": {
    "attributes": {
      "name": "Test Device",
      "ambiId":
"3290458203947453",
      "type": "Ambi::Device",
      "status": "active",
      "signalCapacities": {
        "1": "2.0",
        "2": "1.0",
        "3": "1.0"
      }
    }
  }
}
</pre></td><td colspan="2"><pre>
{
  "data": {
    "id": "2",
    "type": "ambiDevices",
    "attributes": {
      "name": "Test Device",
      "ambiId": "3290458203947453",
      "type": "Ambi::Device",
      "make": null,
      "model": null,
      "serialNumber": null,
      "status": "active",
      "lastHeartbeatAt": "2019-07-04T14:01:13.786+12:00",
      "online": true,
      "signalCapacities": {
        "1": "2.0",
        "2": "1.0",
        "3": "1.0"
      },
      "canSendTestEvent": true,
      "testEventParams": {
        "minLead": 2100,
        "minLength": 900,
        "bufferTime": 1,
        "signals": [
          {
            "id": 5,
            "value": "1"
          },
          {
</pre></td></tr>
</table>

```
            "id": 6,
            "value": "2"
          },
          {
            "id": 7,
            "value": "3"
          }
        ]
      }
    },
    "relationships": {...}
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

### GET/devices/{id}

Get a single device

#### Permissions

Requires the **read_devices** permission for the parent organisation

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /devices/{id} | **Include, filter, fields** | GET | Required |

**Payload**

```
None
```

**Response**

```
{
  "data": {
    "id": "2",
    "type": "ambiDevices",
    "attributes": {
      "name": "Test Device",
      "ambiId": "3290458203947453",
      "type": "Ambi::Device",
      "make": null,
      "model": null,
      "serialNumber": null,
      "status": "active",
      "lastHeartbeatAt": "2019-07-04T14:01:13.786+12:00",
      "online": true,
      "signalCapacities": {
        "1": "2.0",
        "2": "1.0",
        "3": "1.0"
      },
      "canSendTestEvent": true,
      "testEventParams": {
        "minLead": 2100,
        "minLength": 900,
        "bufferTime": 1,
        "signals": [
          {
            "id": 5,
            "value": "1"
          },
          {
            "id": 6,
            "value": "2"
          },
          {
            "id": 7,
            "value": "3"
          }
        ]
      }
    },
    "relationships": {...}
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

### PUT/devices/{id}

Update an existing device

### Permissions

Requires the **write_devices** permission for the parent organisation

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /devices/{id} | **Include, filter, fields** | PUT | Required |

**Payload**

```
{
  "data": {
    "attributes": {
      "name": "Test Device (V2)",
     "signalCapacities": {
        "1": "5.0",
        "2": "10.0",
        "3": "15.0"
      }
    }
  }
}
```

**Response**

```
{
  "data": {
    "id": "2",
    "type": "ambiDevices",
    "attributes": {
      "name": "Test Device (V2)",
      "ambiId": "3290458203947453",
      "type": "Ambi::Device",
      "make": null,
      "model": null,
      "serialNumber": null,
      "status": "active",
      "lastHeartbeatAt": "2019-07-04T14:01:13.786+12:00",
      "online": true,
      "signalCapacities": {
        "1": "5.0",
        "2": "10.0",
        "3": "15.0"
      },
      "canSendTestEvent": true,
      "testEventParams": {
        "minLead": 2100,
        "minLength": 900,
        "bufferTime": 1,
        "signals": [
          {
            "id": 5,
            "value": "1"
          },
          {
            "id": 6,
            "value": "2"
          },
          {
            "id": 7,
            "value": "3"
          }
        ]
      }
    },
    "relationships": {...}
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

## DELETE /devices/{id}

Delete a single device

### Permissions

Requires the **write_devices** permission for the parent organisation

| **Endpoint** | **Query Options** | **Method** | **Authentication** |
|---|---|---|---|
| /devices/{id} | **Include, filter, fields** | DELETE | Required |

| **Payload** | **Response** |
|---|---|
| None | |

```
{
  "data": {
    "id": "2",
    "type": "ambiDevices",
    "attributes": {
      "name": "Test Device",
      "ambiId": "3290458203947453",
      "type": "Ambi::Device",
      "make": null,
      "model": null,
      "serialNumber": null,
      "status": "active",
      "lastHeartbeatAt": "2019-07-04T14:01:13.786+12:00",
      "online": true,
      "signalCapacities": {
        "1": "2.0",
        "2": "1.0",
        "3": "1.0"
      },
      "canSendTestEvent": true,
      "testEventParams": {
        "minLead": 2100,
        "minLength": 900,
        "bufferTime": 1,
        "signals": [
          {
            "id": 5,
            "value": "1"
          },
          {
            "id": 6,
            "value": "2"
          },
          {
            "id": 7,
            "value": "3"
          }
        ]
      }
    },
    "relationships": {...}
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

## Load Types

### GET /load_types

Get a list of all load types

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /load_types | **page[number], page[size]** | GET | Required |

**Payload**

```
None
```

**Response**

```
{
  "data": [
    {
      "id": "1",
      "type": "loadTypes",
      "attributes": {
        "name": "Lighting"
      }
    },
    {
      "id": "2",
      "type": "loadTypes",
      "attributes": {
        "name": "Refrigeration"
      }
    },
    {
      "id": "3",
      "type": "loadTypes",
      "attributes": {
        "name": "HVAC"
      }
    },
    {
      "id": "4",
      "type": "loadTypes",
      "attributes": {
        "name": "Generation"
      }
    }
  ],
  "meta": {
    "totalPages": 1,
    "count": 4,
    "page": 1
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

# Organisations

## GET /organisations/{type}

Get a list of all organisations of a certain type (e.g. Meter Owner, Retailer, Distributor), to use when creating a new site or substation

**type:** should be one of distributor, retailer or meter_owner

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /organisations/{type} | **page[number], page[size]** | GET | Required |

| Payload | Response |
|---|---|
| None | {<br>"data": [<br>{<br>"id": "6",<br>"type": "organisations",<br>"attributes": {<br>"name": "Advanced Metering Solutions"<br>}<br>},<br>{<br>"id": "9",<br>"type": "organisations",<br>"attributes": {<br>"name": "Advanced Metering Services Limited"<br>}<br>},<br>...<br>]<br>} |

## Verification Methods

### GET /verification_methods

Get a list of all verification methods available to be used on a site or substation

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/verification_methods</td><td></td><td>GET</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2"><pre>None</pre></td><td colspan="2"><pre>{
  "data": [
    {
      "id": "1",
      "type": "verificationMethods",
      "attributes": {...},
      "relationships": {...}
    },
    {
      "id": "2",
      "type": "verificationMethods",
      "attributes": {...},
      "relationships": {...}
    },
    ...
  ],
  "meta": {
    "totalPages": 1,
    "count": 3,
    "page": 1
  },
  "jsonapi": {
    "version": "1.0"
  }
}</pre></td></tr>
</table>

## Programmes

### GET /programmes

Get a list of programmes accessible to the current user

<table>
<tr><th>Endpoint</th><th>Query Options</th><th>Method</th><th>Authentication</th></tr>
<tr><td>/programmes</td><td></td><td>GET</td><td>Required</td></tr>
<tr><th colspan="2">Payload</th><th colspan="2">Response</th></tr>
<tr><td colspan="2">None</td><td colspan="2"><pre>
{
  "data": [
    {
      "id": "1",
      "type": "programmes",
      "attributes": {...},
      "relationships": {...}
    },
    {
      "id": "2",
      "type": "programmes",
      "attributes": {...},
      "relationships": {...}
    },
    ...
  ],
  "meta": {
    "totalPages": 1,
    "count": 3,
    "page": 1
  },
  "jsonapi": {
    "version": "1.0"
  }
}
</pre></td></tr>
</table>

## GXPs

### GET /gxps

Get a list of all gxps available to be used on a site or substation

| Endpoint | Query Options | Method | Authentication |
|---|---|---|---|
| /gxps | **filter[name], filter[code]** | GET | Required |

| Payload | Response |
|---|---|
| None | see below |

```
{
  "data": [
    {
      "id": "1",
      "type": "gxps",
      "attributes": {
        "name": "Albury",
        "code": "ABY001"
      },
      "relationships": {...}
    },
    {
      "id": "2",
      "type": "gxps",
      "attributes": {...},
      "relationships": {...}
    },
    ...
  ],
  "meta": {
    "totalPages": 1,
    "count": 3,
    "page": 1
  },
  "jsonapi": {
    "version": "1.0"
  }
}
```

# Swagger Tools

## Swagger API Docs

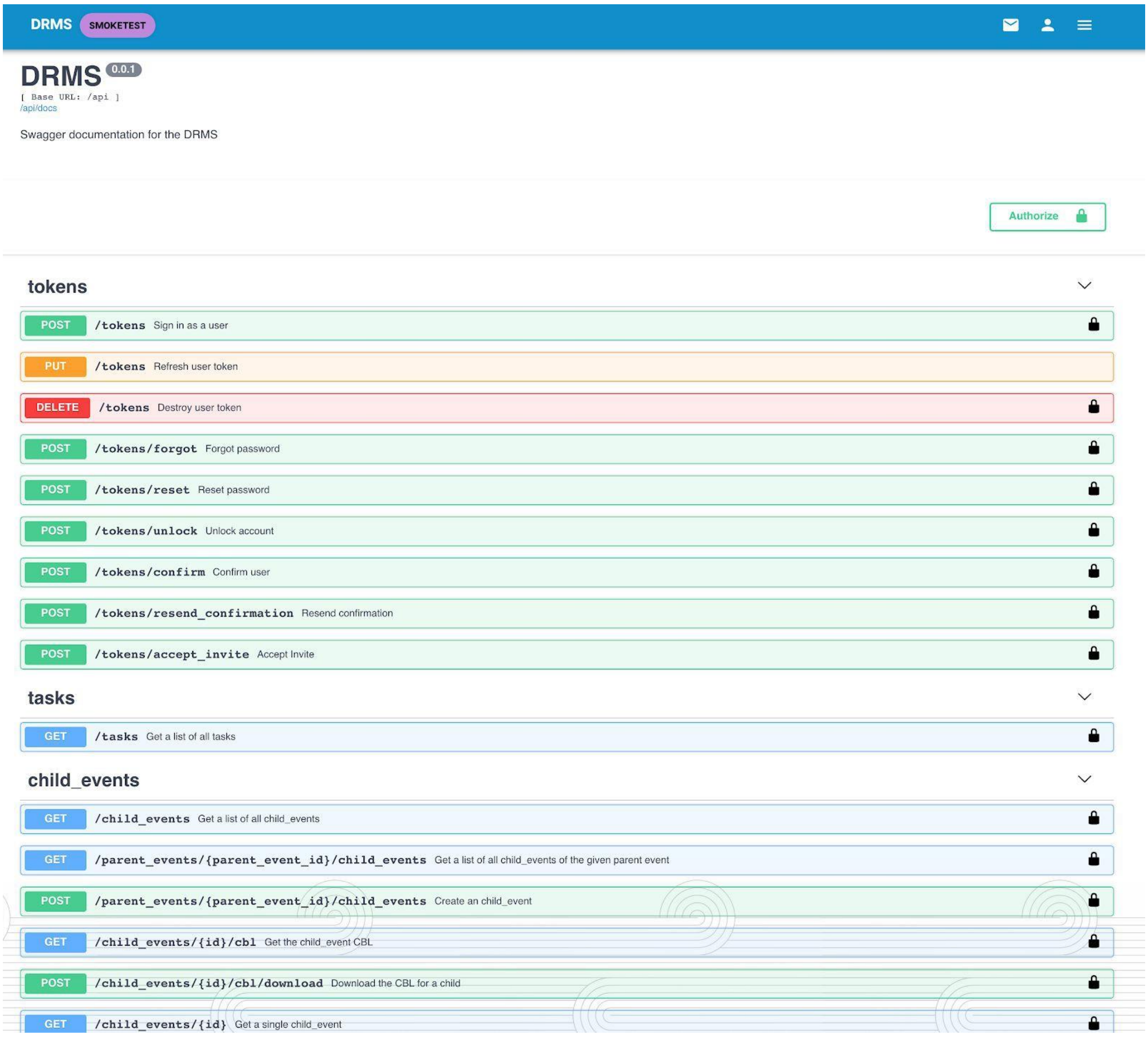
The raw Swagger API documentation for the DERMS Participant API can be found at: https://app.flexpoint.co.nz/api/docs. **Note:** This request **does** require authentication. E.g.

```
curl -H 'Authorization: eyJhbGciOiJIUzI1NiJ9…' -H 'Content-Type:
application/json' https://app.flexpoint.co.nz/api/docs
```

## Swagger UI Playground

An authenticated user can access a Swagger user interface (Swagger UI) from within the DERMS web application. As an authenticated user, visit: https://app.flexpoint.co.nz/apidocs.

E.g. *Screenshot of the Swagger UI playground*



DRMS SMOKETEST

DRMS 0.0.1

[ Base URL: /api ]

/api/docs

Swagger documentation for the DRMS

Authorize

tokens

POST /tokens Sign in as a user

PUT /tokens Refresh user token

DELETE /tokens Destroy user token

POST /tokens/forgot Forgot password

POST /tokens/reset Reset password

POST /tokens/unlock Unlock account

POST /tokens/confirm Confirm user

POST /tokens/resend_confirmation Resend confirmation

POST /tokens/accept_invite Accept Invite

tasks

GET /tasks Get a list of all tasks

child_events

GET /child_events Get a list of all child_events

GET /parent_events/{parent_event_id}/child_events Get a list of all child_events of the given parent event

POST /parent_events/{parent_event_id}/child_events Create an child_event

GET /child_events/{id}/cbl Get the child_event CBL

POST /child_events/{id}/cbl/download Download the CBL for a child

GET /child_events/{id} Get a single child_event

# Appendix 1. HTTP Response Codes

| CODE | MESSAGE |
|---|---|
| 1×× Informational | |
| 100 | 100 Continue |
| 101 | 101 Switching Protocols |
| 102 | 102 Processing |
| 2×× Success | |
| 200 | 200 OK |
| 201 | 201 Created |
| 202 | 202 Accepted |
| 204 | 204 No Content |
| 3×× Redirection | |
| 300 | 300 Multiple Choices |
| 301 | 301 Moved Permanently |
| 302 | 302 Found |
| 303 | 303 See Other |
| 304 | 304 Not Modified |
| 305 | 305 Use Proxy |
| 4×× Client Error | |
| 400 | 400 Bad Request |
| 401 | 401 Unauthorized |
| 404 | 404 Not Found |
| 405 | 405 Method Not Allowed |
| 408 | 408 Request Timeout |
| 429 | 429 Too Many Requests |
| 5×× Server Error | |
| 500 | 500 Internal Server Error |
| 502 | 502 Bad Gateway |
| 503 | 503 Service Unavailable |
| 504 | 504 Gateway Timeout |
| 505 | 505 HTTP Version Not Supported |
| 508 | 508 Loop Detected |