2025

MINT Planning API

VERSION 2.0

EN

Table of Contents

Table of Figures

Table of Revisions

Date	Version	Description of Changes	Status	Author
23-08-2024	1.0	Initial version		Bert Vandormael
04-10-2024	1.1	Remove EquipmentId and add TransactionId to 'plan' endpoint		Bert Vandormael
10-01-2025	2.0	New template	Release	Marco Cicarelli

1 About

This document will describe how an external party can send plans to the MINT platform.

For any other integrations, please get in touch with your local Phoenix Contact subsidiary for the right documentation.

2 The MINT API

The MINT API is an extension of the MINT platform, allowing external parties to send plans for sites and chargers in the field.

3 Getting started

To integrate the MINT Planning API into your application, a client ID and secret need to be requested by the external party.

3.1 Credentials

The necessary credentials to connect to the API can be requested through your local Phoenix Contact subsidiary. In the first stage, credentials for the test environment will be provided. At a later stage, after validation by your subsidiary, production credentials can be provided as well.

Please provide the following when requesting API access:

- Purpose of the integration.
- Overview of the sites for which plans will be sent.

Next, the external party will either ...

- ... receive negative feedback, and no credentials will be provided.
- ... or positive feedback and the necessary credentials and endpoint will be provided.

3.2 Integration

The next step is for the external party to start integrating with the API. Below, every aspect of the API will be described. Read them carefully, since all of them will be important to set up a working integration.

3.2.1 Environment

As mentioned in section 3.1, depending on the environment a request is made for, the correct endpoint will be provided.

Each environment makes use of versioning, whenever there is a breaking change, an updated version will have to be used in the URL. This versioning is always placed before the endpoint, e.g. /api/v1.0/....

3.2.2 Security

To use any of the endpoints on the API, an access token is required. This token can be requested using the authentication endpoint.

POST /auth/token

Request body

```
{
        "clientId": "<YOUR CLIENT ID>",
        "clientSecret": "<YOUR CLIENT SECRET>"
}
```

Responses

200 OK

```
{
        "accessToken": "<YOUR ACCESS TOKEN>",
        "expiration": "0000-00-00T00:00:00Z"
}
```

401 Unauthorized

```
<empty response>
```

3.2.3 Validation

When a validation error occurs on an endpoint, the validation error will be returned in the following format:

400 Bad Request

```
[
        {
                "message": "Validation error 1"
        },
        {
                "message": "Validation error 2"
        }
        ...
]
```

3.2.4 Health

A health endpoint is available to check the status of the API.

GET /health/

Responses

200 OK

```
<empty response>
```

503 Service Unavailable

```
<empty response>
```

3.2.5 The “status” field

Every endpoint has a "status" field defined. It gives the third party the option to send additional, custom information. In a standard API implementation, this field will not be processed by either the MINT DataHub or the PLC. Though, we strongly encourage to make use of this field. For logging, tracing or any other purpose that would have use of this information. In case you decide not to provide a status, the field can be left empty.

4 Endpoints

The API provides the following endpoints. For details on authentication, please refer to section 3.2.2.

4.1 Plan

POST /plan/

Headers

```
Authorization: Bearer <YOUR ACCESS TOKEN>
```

Request body

```
{
        "messageId": "UUID",
        "locationId": "string",
        "transactionId": "UUID",
        "status": "string",
        "timestamp": "0000-00-00T00:00:00Z",
        "scheduledStartTime": "0000-00-00T00:00:00Z",
        "scheduledEndTime": "0000-00-00T00:00:00Z",
                // Use either expectedRange with averageConsumption
        "expectedRange": 0,                                     // in km
        "averageConsumption": 0.0,                              // in kWh/100km, optional
                // or expectedCharge
        "expectedCharge": 0,                                    // in kWh
}
```

Table 1: Plan

Responses

200 OK

```
<empty response>
```

400 Bad Request

See validation in section 3.2.3.

401 Unauthorized

```
<empty response>
```