Data API Whitepaper

Author: Hepting Manuel (BT-VS/XSW-AIA)

Date: 7 October, 2021

1 Introduction

This document describes the Intelligent Insights Data API available in Intelligent Insights 1.0.2.

The Intelligent Insights API interface is a RESTful API interface designed as an application-to-application interface in order to pull data from the system. The API is defined according to the OpenAPI 3.0 specifications.

You enable the API Gateway with the IGI-XDataAPI license in the software. Once a valid license is available on the system, the system admin can add an API key to the system. The API key is used to authenticate in the Intelligent Insights system when connecting to the system by API. In version 1.0.2, you can only add one API key to the system.

The Intelligent Insights API allows you to perform actions on various resources available in a RESTful manner. In order to query data from the Intelligent Insights software, the API offers "GET" actions on two types of endpoints, the configuration endpoint and the data endpoint. The configuration endpoint provides the available data sources. The data API endpoint provides the collected data of the available sources. Available data sources are the counter tasks, the occupancy counting tasks and the crowd detection fields of cameras, or the fill level of areas. When running GET queries on data sources, the query has to include the desired time interval, granularity and the data sources. The result of the collected data is provided in JSON format.

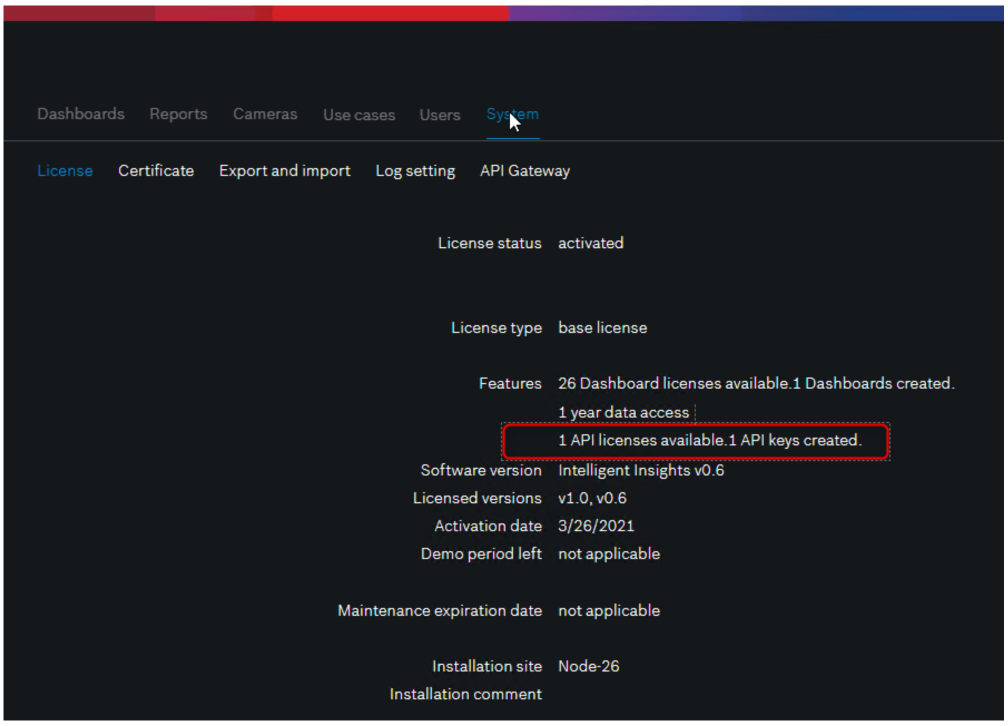
The API is not designed to get real time data from the Intelligent Insights system. With the API Gateway, a 3rd party API can query data from the system maximum once in a minute. Only one active query is allowed. You can only start a second query if the previous query finished.

One use case example of the API Gateway is that the occupancy or the area fill of the entire day is queried once a day from the system and stored in a 3rd party system. A second use case example is that you generate a customized dashboard with different data visualization. You can not update the data in real time. You can only execute maximum one query per minute.

Note: The limitation of one query per minute is required to ensure the live data streaming in the Intelligent Insights dashboards and the data insertion to the database.

1.1 API licensing

In order to use the API interface, you must provide a valid license for the Data API. Order the IGI-XDATAAPI license and activate the license for your system.



License status	activated
License type	base license
Features	26 Dashboard licenses available.1 Dashboards created.
	1 year data access
	1 API licenses available.1 API keys created.
Software version	Intelligent Insights v0.6
Licensed versions	v1.0, v0.6
Activation date	3/26/2021
Demo period left	not applicable
Maintenance expiration date	not applicable
Installation site	Node-26
Installation comment	

1.2 Certificate handling

The API interface uses the latest TLS 1.3 encryption and requires a valid certificate of the Intelligent Insights server on the hostmachine, from which you want to send the API request to the Intelligent Insights server.

How to create or import a certificate in the Intelligent Insights system, refer here.

If you want to use the Intelligent Insights API with tools like Postman, follow the tool requirements to install the certificate. When using the Intelligent Insights API Gateway by scripting, the certificate installation process and place of the certificate depends on the library you are using. Follow the recommendation of the library.

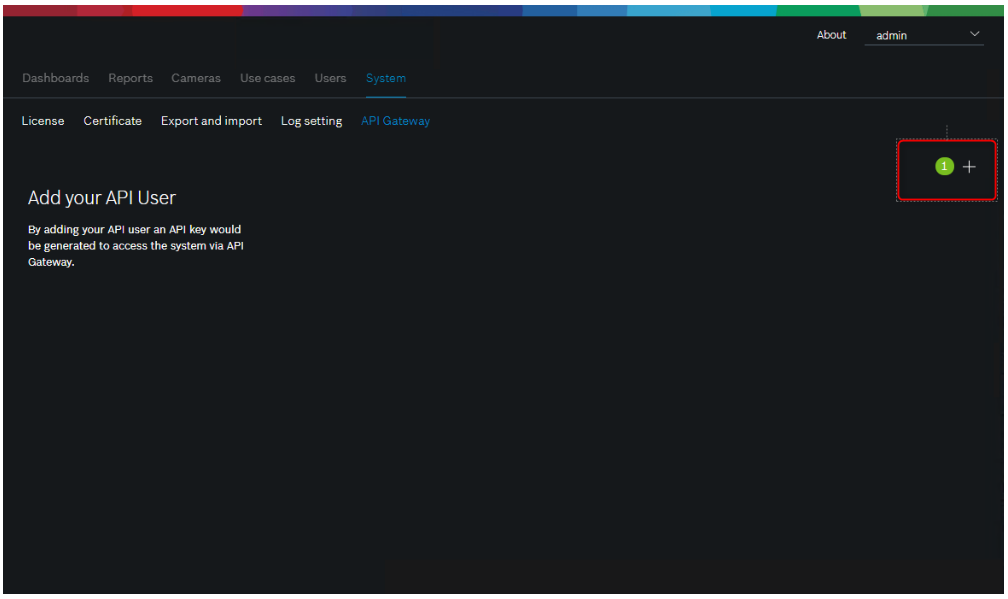
1.3 Authentication

The Intelligent Insights API interface uses key based authentication. When adding the API user, the system generates a unique value. User can not change the API key. You have to provide the API key with every API request to the Intelligent Insights system.

Note: Adding an API user is only possible once a valid license is available on the Intelligent Insights system.

Adding an API user:

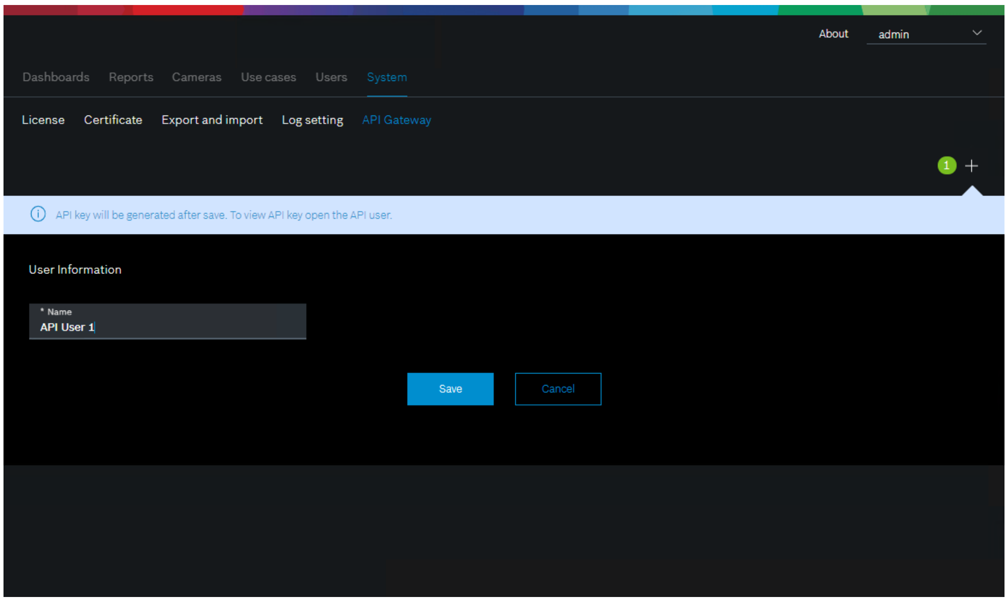
1. Go to **System** > **API Gateway** and click the **+** button.



Add your API User

By adding your API user an API key would be generated to access the system via API Gateway.

2. Enter a user name for the API user. The API key for the authentication is generated automatically.

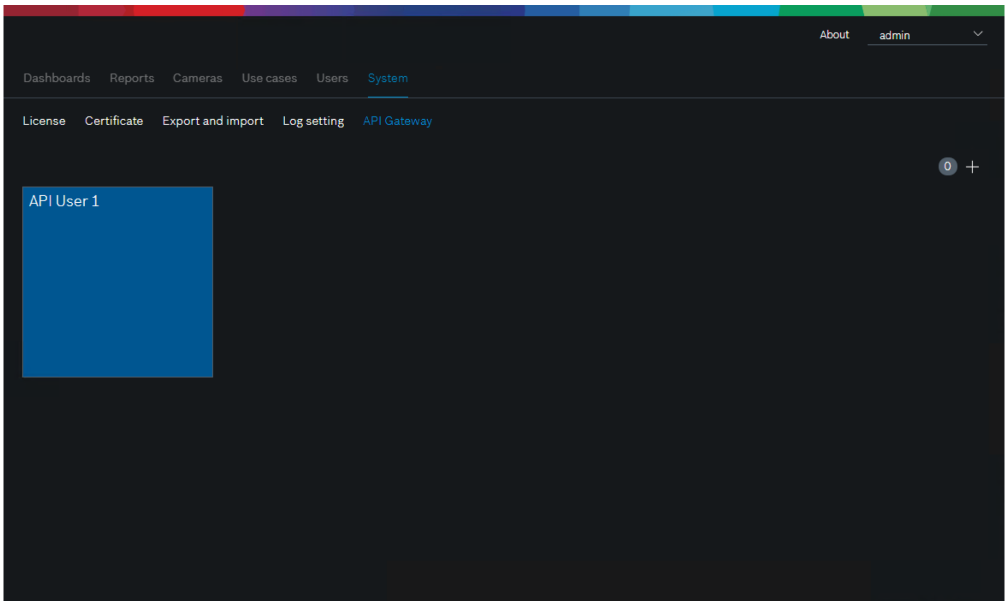


API key will be generated after save. To view API key open the API user.

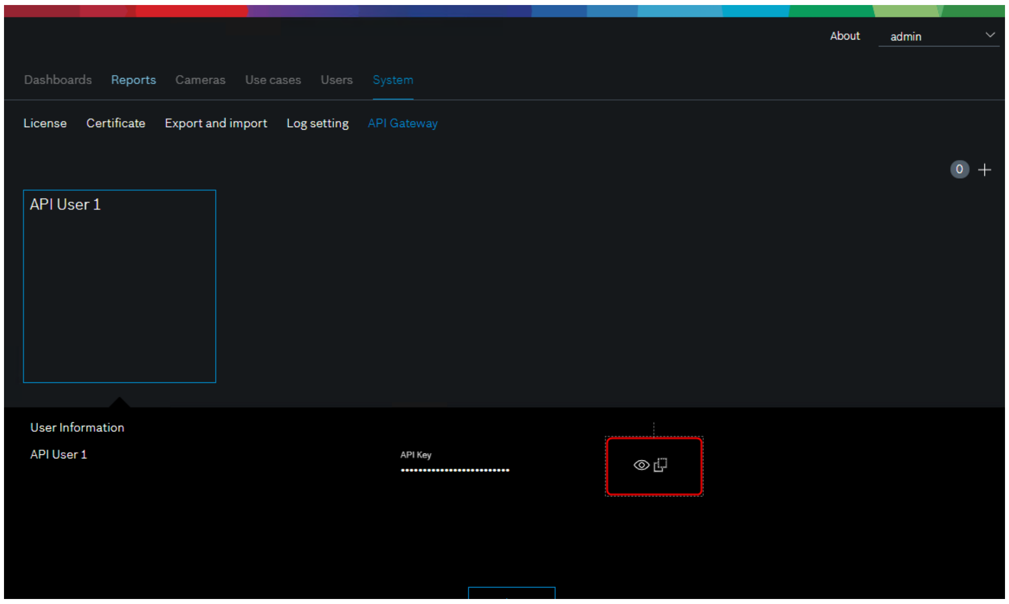
User Information

* Name
API User 1

3. Click **Save**.



4. In order to review the API key, click on the API user name and either view or copy the API key:



1.4 Documentation and testing the API interface

For API documentation and testing purpose, Intelligent Insights offers a Swagger interface.

Swagger is an interface description language for describing RESTful APIs using JSON. Swagger is used together with a set of open-source software tools to design, build, document and use RESTful web services. Swagger includes automated documentation, code generation (into many programming languages), and test case generation.

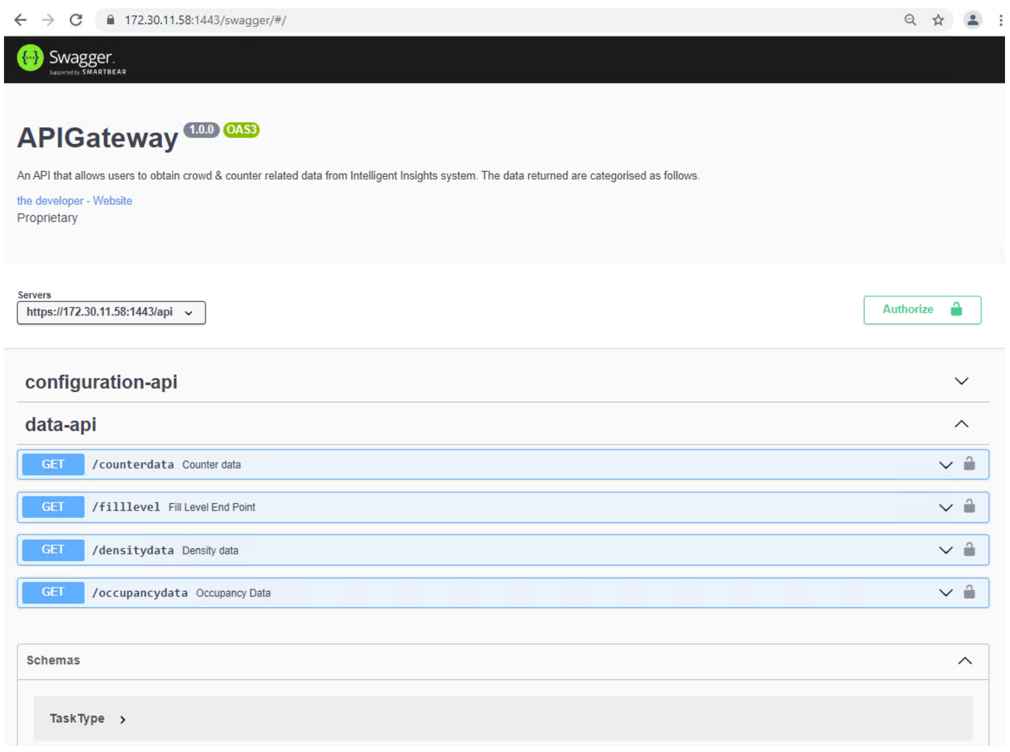
In order to access the Swagger interface, click on the link in the API Gateway tab or open the URL **https:\\ipaddress:port\Swagger**.

The standard port of the API interface is 1443. You can adjust the port in the installation process of Intelligent Insights 1.0.2.

1.4.1 Authentication on the swagger interface

Access the Swagger interface:

1. Open the URL **https:\\ipaddress:port\Swagger**.



APIGateway 1.0.0 OAS3

An API that allows users to obtain crowd & counter related data from Intelligent Insights system. The data returned are categorised as follows.

the developer - Website
Proprietary

Servers
https://172.30.11.58:1443/api

configuration-api

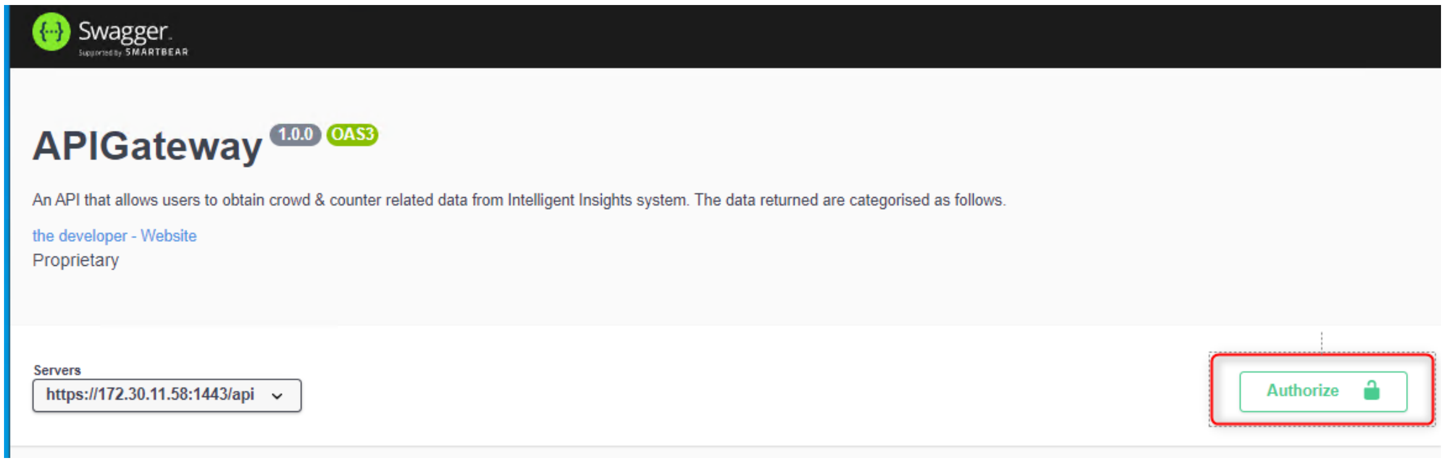
data-api

GET /counterdata Counter data
GET /filllevel Fill Level End Point
GET /densitydata Density data
GET /occupancydata Occupancy Data

Schemas

TaskType

2. Enter the API key in order to authorize on the Intelligent Insights API gateway:



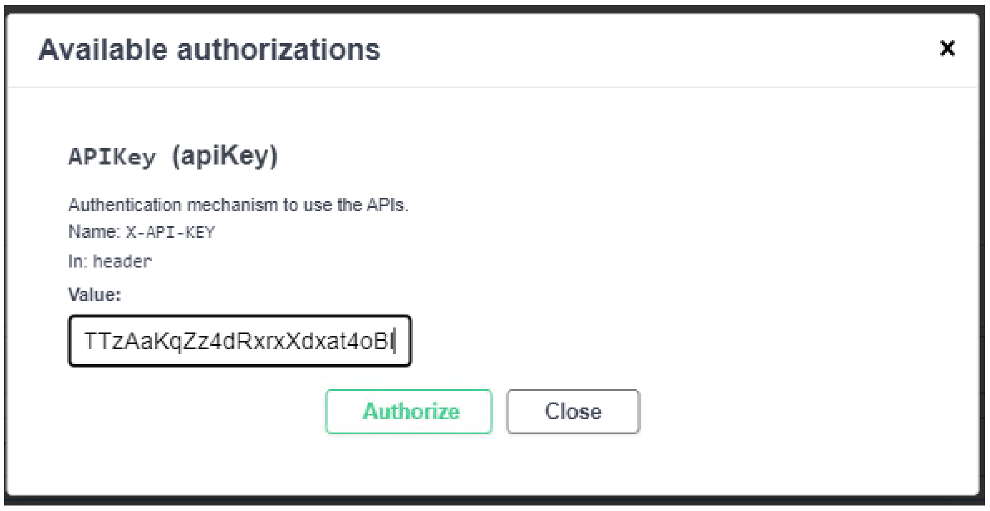
APIGateway 1.0.0 OAS3

An API that allows users to obtain crowd & counter related data from Intelligent Insights system. The data returned are categorised as follows.

the developer - Website
Proprietary

Servers
https://172.30.11.58:1443/api

3. Enter the API copied from the Intelligent Insights API user (see section "Authentication").

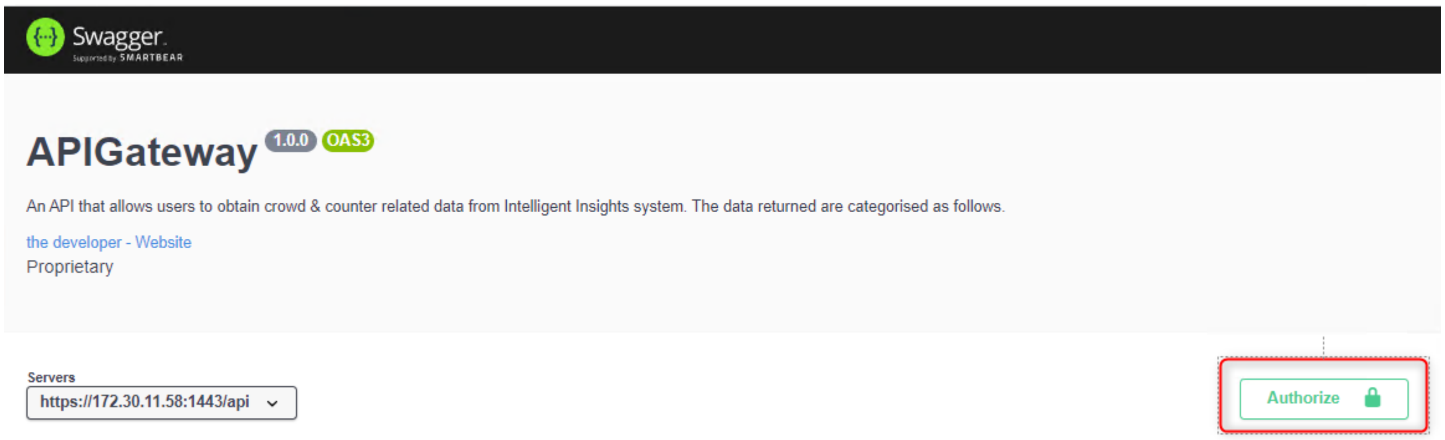


Available authorizations

APIKey (apiKey)

Authentication mechanism to use the APIs.
Name: X-API-KEY
In: header
Value:
TTzAaKqZz4dRxrxXdxat4oBl

4. Authenticate with the API user:



APIGateway 1.0.0 OAS3

An API that allows users to obtain crowd & counter related data from Intelligent Insights system. The data returned are categorised as follows.

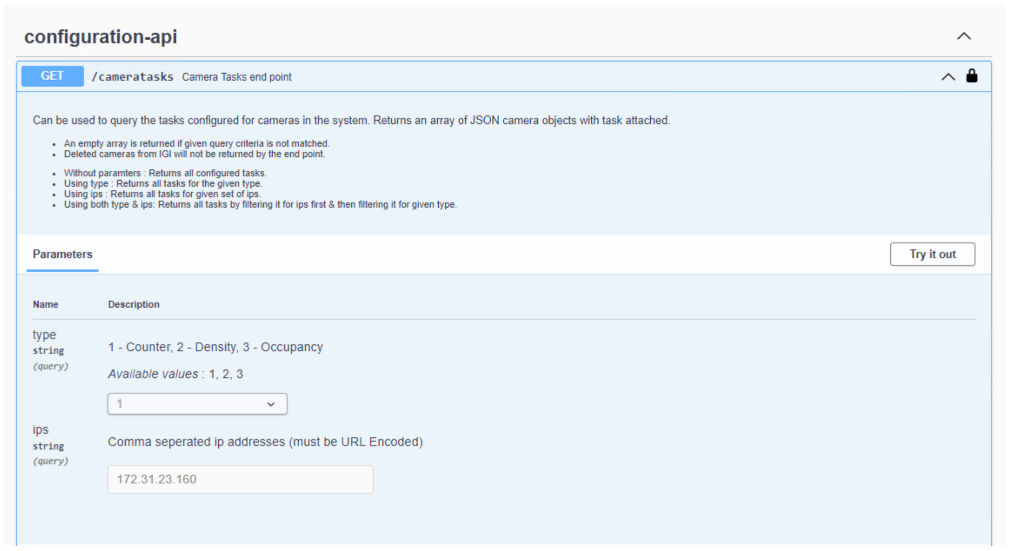
the developer - Website
Proprietary

Servers
https://172.30.11.58:1443/api

1.4.2 API documentation in Swagger

The swagger interface provides a detailed API description about the parameters and how to use the parameters.

Example: Configuration API



configuration-api

GET `/cameratasks` Camera Tasks end point

Can be used to query the tasks configured for cameras in the system. Returns an array of JSON camera objects with task attached.

- An empty array is returned if given query criteria is not matched.
- Deleted cameras from IGI will not be returned by the end point.

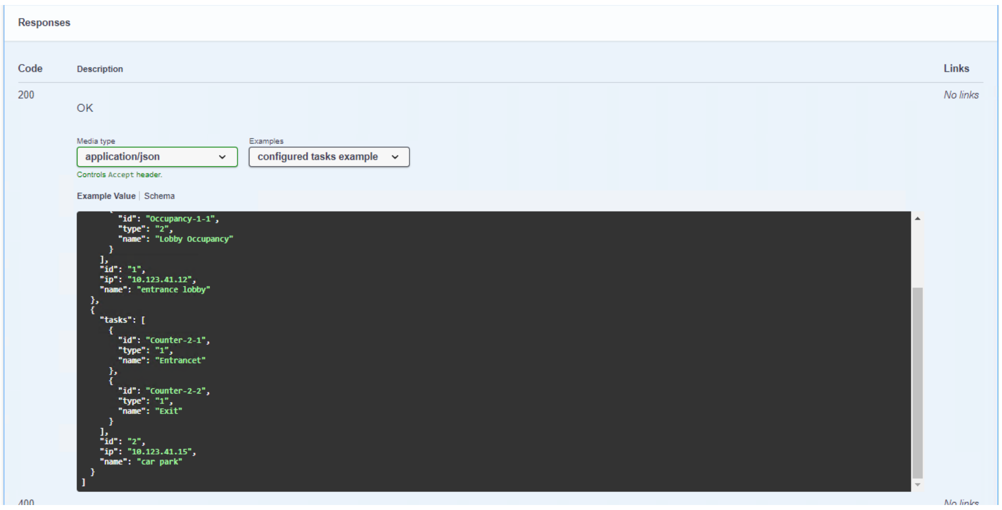
- Without paramters : Returns all configured tasks.
- Using type : Returns all tasks for the given type.
- Using ips : Returns all tasks for given set of ips.
- Using both type & ips: Returns all tasks by filtering it for ips first & then filtering it for given type.

Parameters

Name	Description
type string *(query)*	1 - Counter, 2 - Density, 3 - Occupancy *Available values* : 1, 2, 3 1
ips string *(query)*	Comma seperated ip addresses (must be URL Encoded) 172.31.23.160

API Response

In general, if a request on a resource is successful, an HTTP status code 200 returns and the API response is provided in JSON format:


Responses

Code | Description | Links

200 OK | No links

Media type: application/json — Controls Accept header.
Examples: configured tasks example

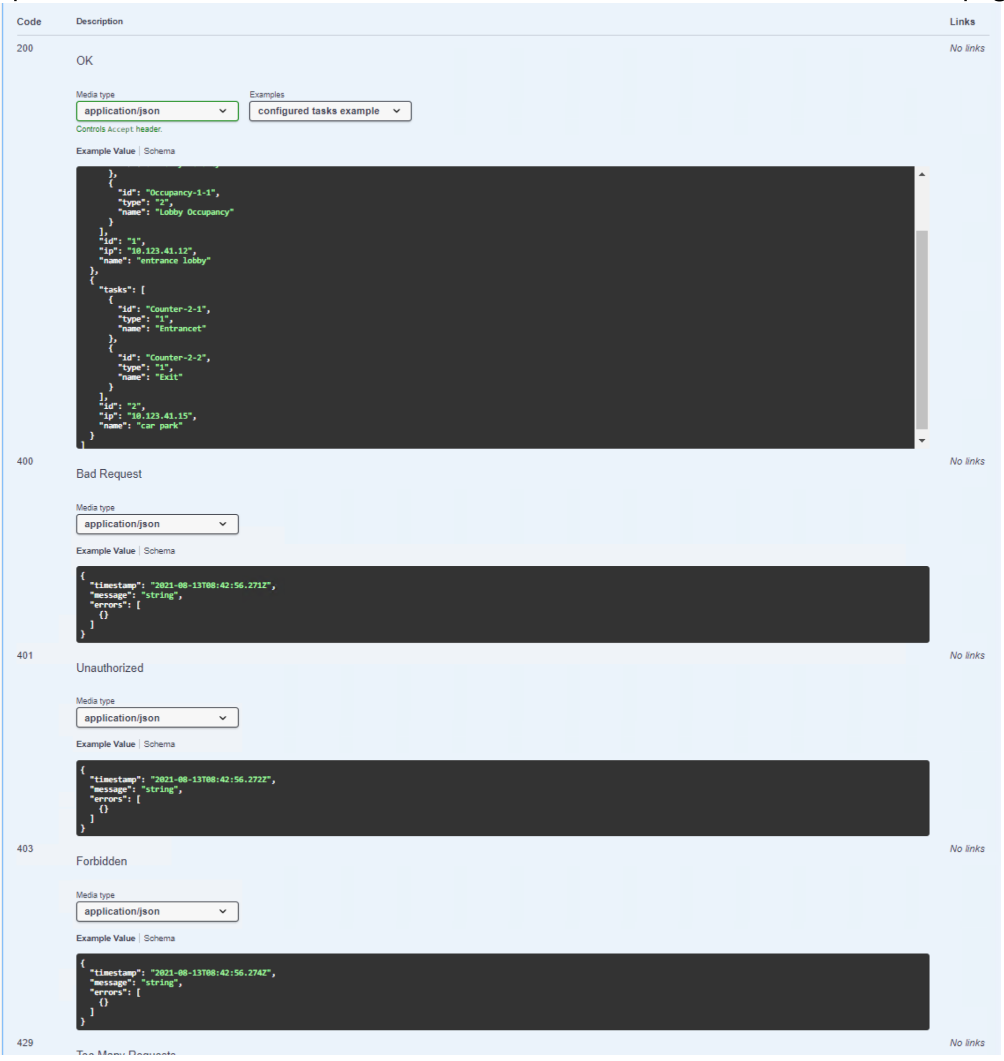
Example Value | Schema

```
        "id": "Occupancy-1-1",
        "type": "2",
        "name": "Lobby Occupancy"
      }
    ],
    "id": "1",
    "ip": "10.123.41.12",
    "name": "entrance lobby"
  },
  {
    "tasks": [
      {
        "id": "Counter-2-1",
        "type": "1",
        "name": "Entrancet"
      },
      {
        "id": "Counter-2-2",
        "type": "1",
        "name": "Exit"
      }
    ],
    "id": "2",
    "ip": "10.123.41.15",
    "name": "car park"
  }
]
```

400 No links

If a request fails, the error information returns with the HTTP status code along with a more detailed explanation JSON format (depending on the request).

Ypi find a list of error codes and their associated HTTP status codes on the errors page:


Code | Description | Links

200 OK | No links

Media type: application/json — Controls Accept header.
Examples: configured tasks example

Example Value | Schema

```
      },
      {
        "id": "Occupancy-1-1",
        "type": "2",
        "name": "Lobby Occupancy"
      }
    ],
    "id": "1",
    "ip": "10.123.41.12",
    "name": "entrance lobby"
  },
  {
    "tasks": [
      {
        "id": "Counter-2-1",
        "type": "1",
        "name": "Entrancet"
      },
      {
        "id": "Counter-2-2",
        "type": "1",
        "name": "Exit"
      }
    ],
    "id": "2",
    "ip": "10.123.41.15",
    "name": "car park"
  }
]
```

400 Bad Request | No links

Media type: application/json

Example Value | Schema

```
{
  "timestamp": "2021-08-13T08:42:56.271Z",
  "message": "string",
  "errors": [
    {}
  ]
}
```

401 Unauthorized | No links

Media type: application/json

Example Value | Schema

```
{
  "timestamp": "2021-08-13T08:42:56.272Z",
  "message": "string",
  "errors": [
    {}
  ]
}
```

403 Forbidden | No links

Media type: application/json

Example Value | Schema

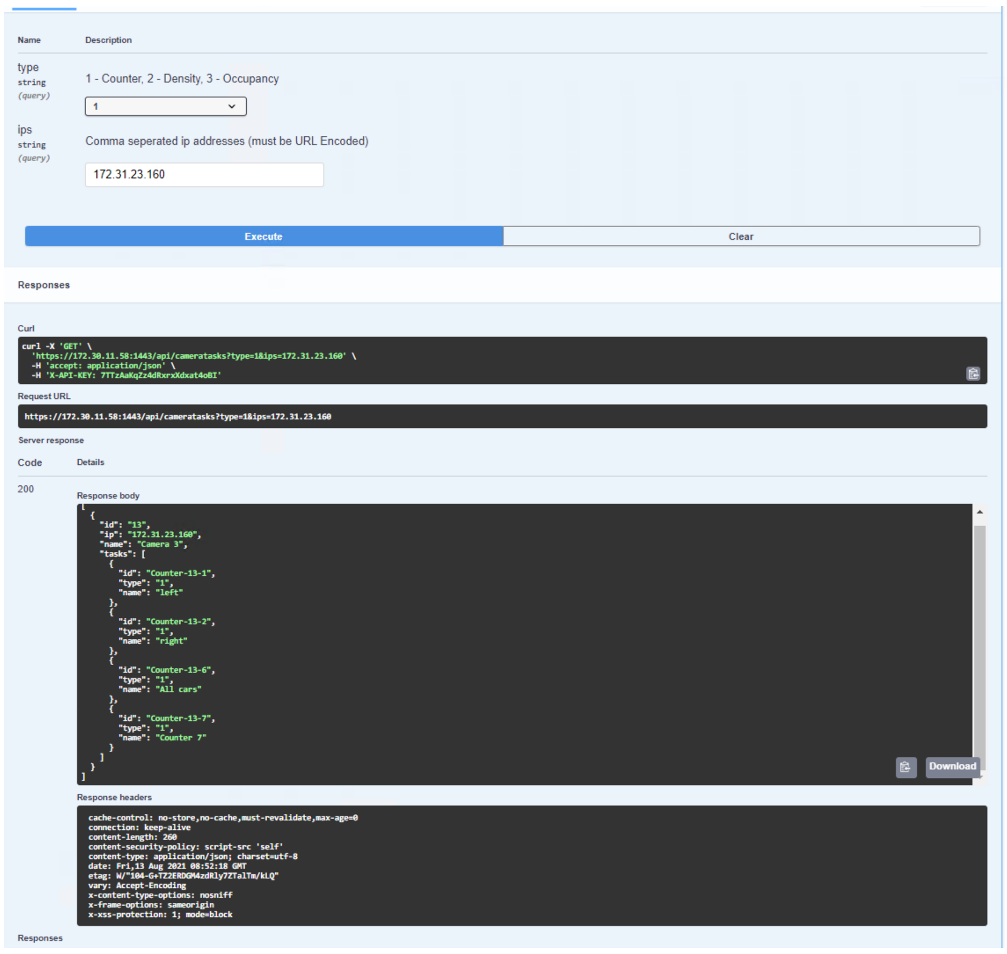
```
{
  "timestamp": "2021-08-13T08:42:56.274Z",
  "message": "string",
  "errors": [
    {}
  ]
}
```

429 Too Many Requests | No links

1.4.3 API testing

With the **Try it out** button, the Swagger interface allows to query data from the system and to copy the tested URL.

Example with the configuration API and camera 172.31.23.160:



Name	Description
type string *(query)*	1 - Counter, 2 - Density, 3 - Occupancy 1
ips string *(query)*	Comma seperated ip addresses (must be URL Encoded) 172.31.23.160

Curl

```
curl -X 'GET' \
  'https://172.30.11.58:1443/api/cameratasks?type=1&ips=172.31.23.160' \
  -H 'accept: application/json' \
  -H 'X-API-KEY: 7TTzAaKqZz4dRxrxXdxat4oBI'
```

Request URL

```
https://172.30.11.58:1443/api/cameratasks?type=1&ips=172.31.23.160
```

Server response

Code: 200

Response body

```
[
  {
    "id": "13",
    "ip": "172.31.23.160",
    "name": "Camera 3",
    "tasks": [
      {
        "id": "Counter-13-1",
        "type": "1",
        "name": "left"
      },
      {
        "id": "Counter-13-2",
        "type": "1",
        "name": "right"
      },
      {
        "id": "Counter-13-6",
        "type": "1",
        "name": "All cars"
      },
      {
        "id": "Counter-13-7",
        "type": "1",
        "name": "Counter 7"
      }
    ]
  }
]
```

Response headers

```
cache-control: no-store,no-cache,must-revalidate,max-age=0
connection: keep-alive
content-length: 260
content-security-policy: script-src 'self'
content-type: application/json; charset=utf-8
date: Fri,13 Aug 2021 08:52:18 GMT
etag: W/"104-G+TZ2ERDGM4zdRly7ZTalTm/kLQ"
vary: Accept-Encoding
x-content-type-options: nosniff
x-frame-options: sameorigin
x-xss-protection: 1; mode=block
```

When executing the call, the response of the call displays in the response body and you can copy the created request URL.

1.5 API rate limits

There is a rate limit of maximum one parallel API call to the Intelligent Insights system and an API blocking time of 60 seconds.

This means that a 3rd party application can only send one API request to the system per minute. If the API call can be answered within one minute, the system allows to send another call after one minute. If the call can not be answered within one minute because of the requested amount of data, the API Gateway blocks for further calls until the API call finishes.

Intelligent Insights 1.0.2 allows only to configure one API key. Connecting with multiple API keys in parallel is not supported.