

API documentation v1-preview2

Base URL

The base URL used for all APIs is formatted as the following:

```
https://{environment}/v1-preview2
```

REST API documentation via Swagger

IdIssuer now includes the default Swagger UI for displaying the API documentation in a nice human readable way. To browse that UI just open your browser and go to <https://idi-api.pulab.pl/v1-preview2/docs>.

Headers

A typical HTTP request to the Platform API includes the following headers:

```
Content-Type: application/json, charset=utf8
Authorization: Bearer {API_Token}
```

- **Content-Type:** every request must include a `Content-Type` header.
- **API Token:** an API token is required for IdIssuer to identify and authenticate your IdIssuer user.

Authentication

Obtaining an Acces Token

To begin an authorization process flow, you have to have registered user on <https://idissuer.pulab.pl>. If you have an user, you must exchange it for an Access Token that can be used to call IdIssuer API. Using user credentials you will need to `POST` to the token endpoint:

```
https://{environment}/v1-preview2/auth/token
```

Request body

Required	Type	Description
username	string	User login email
password	string	User password

```

POST {{apiHost}}/auth/token HTTP/1.1
content-type: application/json
{
  "username":"{{username}}",
  "password":"{{password}}"
}

```

The response contains the Access Token and Refresh Token

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
{
  "access_token": {
    "token_type": "bearer",
    "token": "eyJhbGciOiJIUzI1NiIsInR... ",
    "expires_in": 3600
  },
  "refresh_token": "5QhI6iaHkyh7Z3eNwhNxcbF2..."
}

```

Refresh Token Workflow

When the user make a request to a protected resources with an expired token, then receives a response 401 unauthorized with header Token-Expired

Sample Response:

```

HTTP/1.1 401 Unauthorized
Content-Length: 0
WWW-Authenticate: Bearer error="invalid_token", error_description="The token is expired"
Token-Expired: true

```

For exchange an refresh token to new Access Token, you will need to POST to the token refresh endpoint:

https://{environment}/v1-preview2/auth/refresh_token

Request body

Required	Type	Description
access_token	string	Access token
refresh_token	string	Refresh token

```

POST {{apiHost}}/auth/token HTTP/1.1
content-type: application/json
{
  "access_token" : "{{authToken}}",
  "refresh_token" : "{{refreshToken}}"
}

```

The response contains the Access Token and Refresh Token

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
{
  "access_token": {
    "token_type": "bearer",
    "token": "eyJhbGciOiJIUzI1NiIsInR5cGU6IjY0bnVudC54b290IiwiaWF0IjoiMTUxMjM0NTY3ODk0In0",
    "expires_in": 3600
  },
  "refresh_token": "5QhI6iaHkyh7Z3eNwhNxcbF2..."
}
```

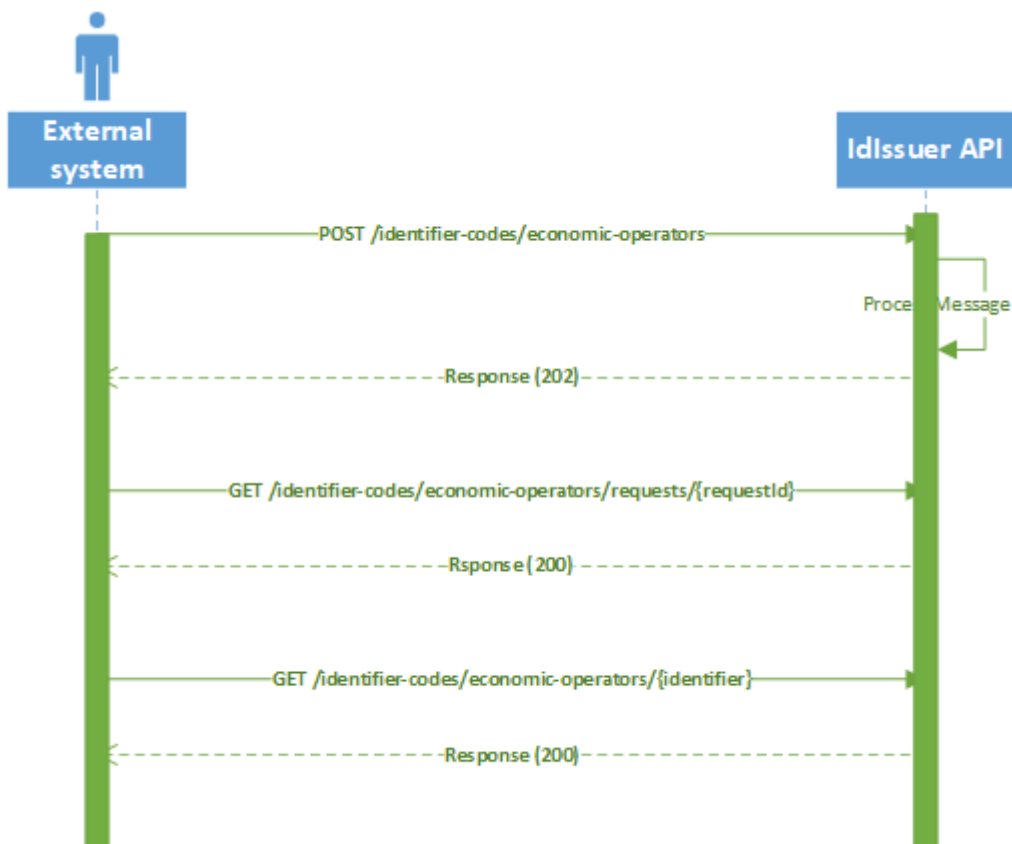
Using an API Token

To use the IdIssuer API with a specific application, you must authenticate a request using your API token. As stated above, an API token must be included in your **HTTP Request Header** for authentication.

```
Athorization: Bearer {API_Token}
```

Registry objects registration flow

Register economic operator



Data example

```
POST {{apiHost}}/identifier-codes/economic-operators/  
Content-Type: application/json  
Authorization: Bearer {{AUTH_TOKEN}}
```

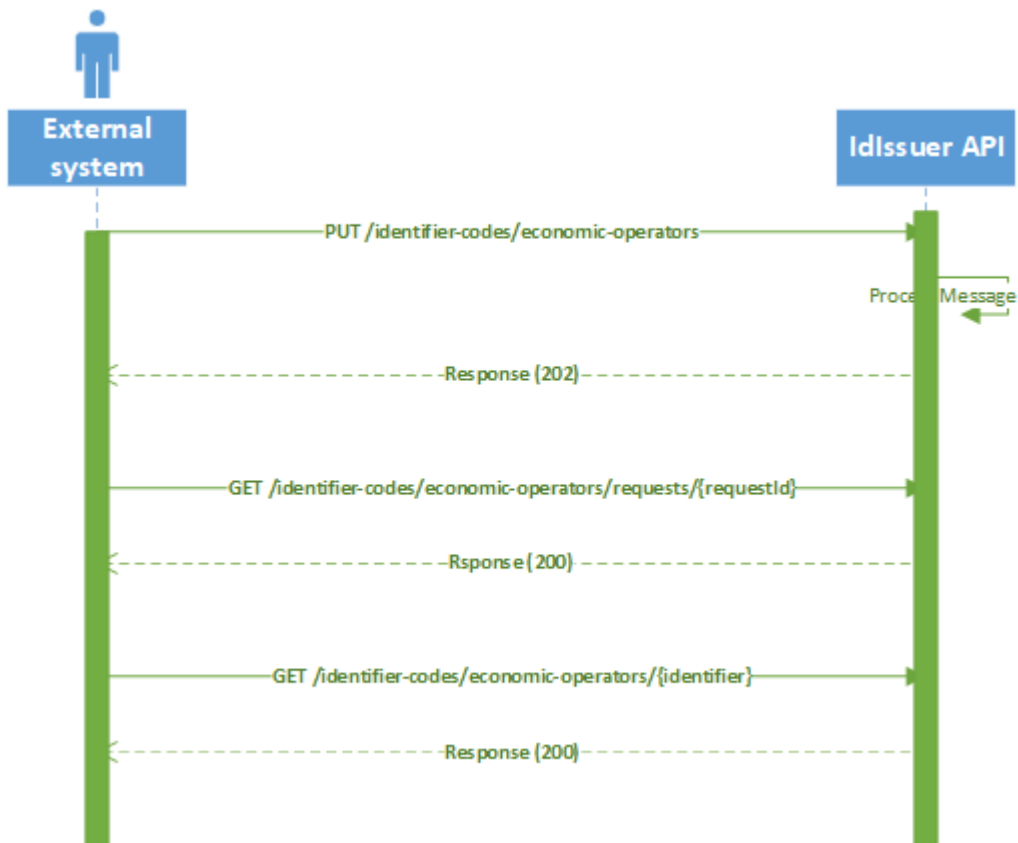
```
{  
  "EO_Name1": "International Cigarette Company",  
  "EO_Name2": "I.C.C. Inc",  
  "EO_Address": "01-123 Warsaw, 1 Romana Sanguszki Street, Mazowieckie",  
  "EO_CountryReg": "PL",  
  "EO_Email": "EconomicOperatorExample@pwpw.pl",  
  "VAT_R": true,  
  "VAT_N": "PL123456789",  
  "TAX_N": "0123456789",  
  "EO_ExciseNumber1": true,  
  "EO_ExciseNumber2": "PL123456789AB",  
  "OtherEOID_R": true,  
  "OtherEOID_N": [  
    "OtherCode12345", "SecondCode12345"  
  ],  
  "Reg_3RD": true,  
  "Reg_EOID": "PLPWPW0000001"  
}
```

Success Response

```
HTTP/1.1 202 Accepted  
Connection: close  
Date: Tue, 05 Mar 2019 09:04:03 GMT  
Content-Type: application/json; charset=utf-8
```

```
{  
  "Request_Id": 681  
}
```

Update economic operator



Data example

```

PUT {{apiHost}}/identifier-codes/economic-operators/
Content-Type: application/json
Authorization: Bearer {{AUTH_TOKEN}}
  
```

```

{
  "EO_ID": "PLPWPW000002b",
  "EO_CODE": "BuIfk107",
  "EO_Name1": "International Cigarette Company",
  "EO_Name2": "I.C.C. Inc",
  "EO_Address": "01-123 Warsaw, 1 Romana Sanguszki Street, Mazowieckie",
  "EO_CountryReg": "PL",
  "EO_Email": "EconomicOperatorExample@pwpw.pl",
  "VAT_R": true,
  "VAT_N": "PL123456789",
  "TAX_N": "0123456789",
  "EO_ExciseNumber1": true,
  "EO_ExciseNumber2": "PL123456789AB",
  "OtherEOID_R": true,
  "OtherEOID_N": [
    "OtherCode12345, SecondCode12345"
  ],
  "Reg_3RD": true,
  "Reg_EOID": "PLPWPW0000001"
}
  
```

Success Response

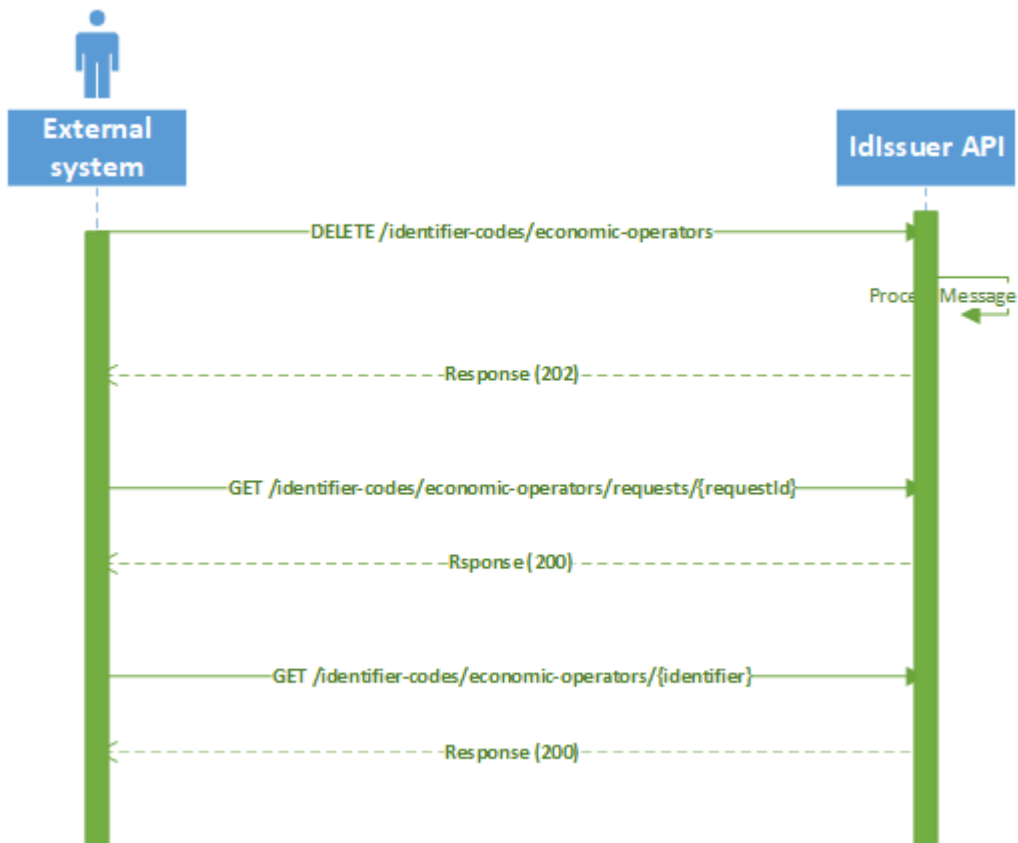
```

HTTP/1.1 202 Accepted
Connection: close
  
```

Date: Tue, 05 Mar 2019 09:04:03 GMT
Content-Type: application/json; charset=utf-8

```
{  
  "Request_Id": 681  
}
```

Deactivate economic operator



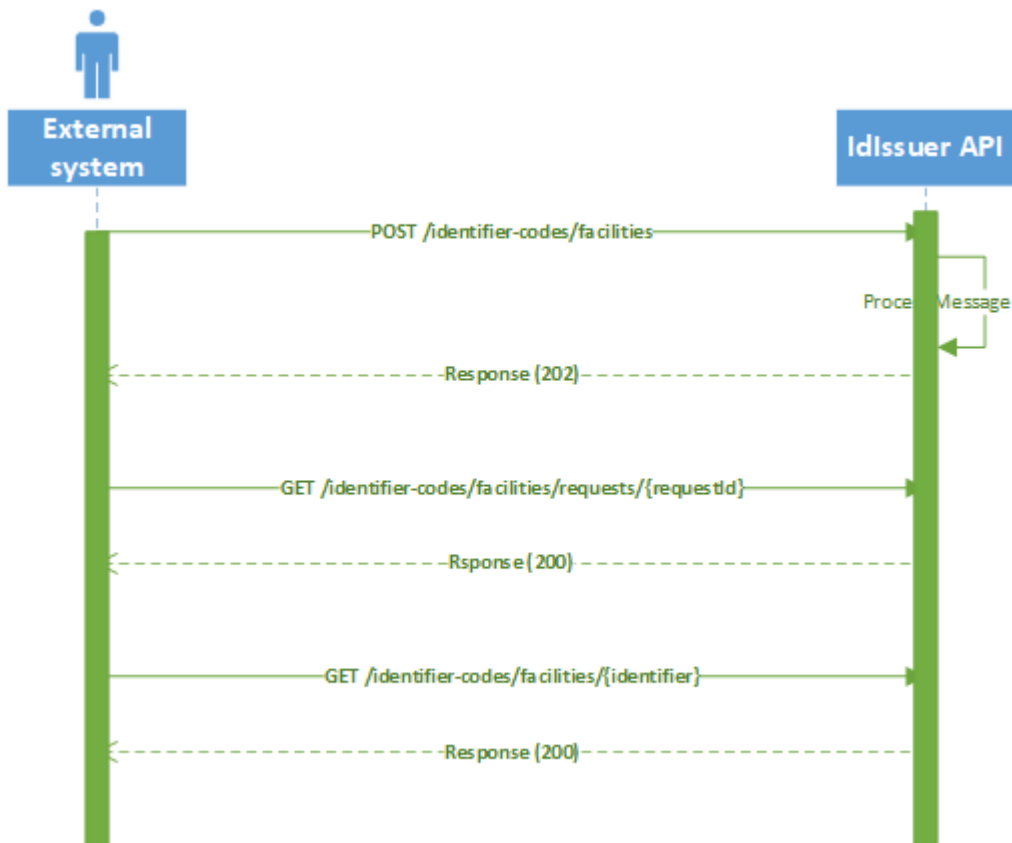
Data example

```
DELETE {{apiHost}}/identifier-codes/economic-operators/  
Content-Type: application/json  
Authorization: Bearer {{AUTH_TOKEN}}  
  
{  
  "EO_ID": "PLPWPWA1B210A",  
  "EO_CODE": "Ex4Mp13",  
  "Reg_3RD": true,  
  "Reg_EOID": "PLPWPW000001"  
}
```

Success Response

```
HTTP/1.1 202 Accepted  
Connection: close  
Date: Tue, 05 Mar 2019 09:04:03 GMT  
Content-Type: application/json; charset=utf-8  
  
{  
  "Request_Id": 681  
}
```

Register facility



Data example

```
POST {{apiHost}}/identifier-codes/facilities/
Content-Type: application/json
Authorization: Bearer {{AUTH_TOKEN}}

{
  "EO_ID": "PLPWPWA1B210A",
  "EO_CODE": "Ex4Mp13",
  "F_Address": "01-123 Warsaw, 10 Romana Sanguszki Street, Mazowieckie",
  "F_Country": "PL",
  "F_Type": 3,
  "F_Type_Other": "string",
  "F_Status": true,
  "F_ExciseNumber1": true,
  "F_ExciseNumber2": "PL123456789AA",
  "OtherFID_R": true,
  "OtherFID_N": [
    "ExampleFacility001"
  ],
  "Reg_3RD": true,
  "Reg_EOID": "PLPWPW0000001"
}
```

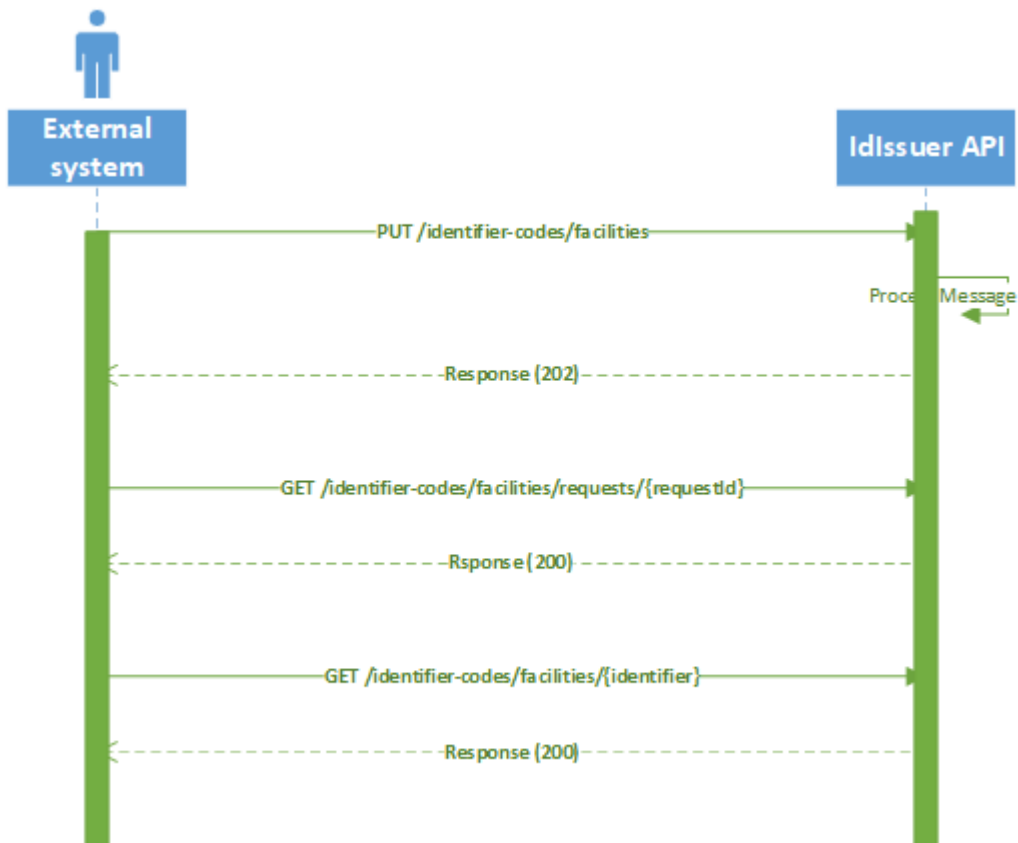
Success Response

```
HTTP/1.1 202 Accepted
Connection: close
Date: Tue, 05 Mar 2019 09:04:03 GMT
Content-Type: application/json; charset=utf-8
```



```
{  
  "Request_Id": 681  
}
```

Update facility



Data example

```
PUT {{apiHost}}/identifier-codes/facilities/  
Content-Type: application/json  
Authorization: Bearer {{AUTH_TOKEN}}
```

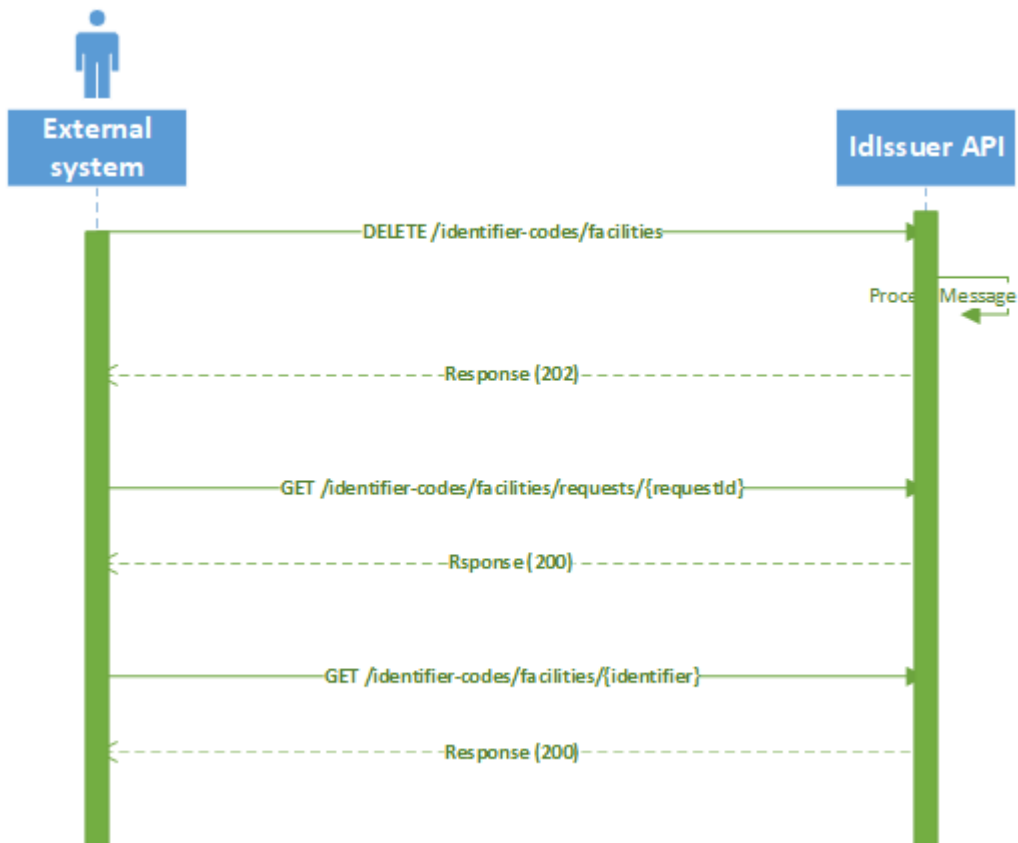
```
{  
  "F_ID": "PLPWPW987ABC",  
  "EO_ID": "PLPWPWA1B210A",  
  "EO_CODE": "Ex4Mp13",  
  "F_Address": "01-123 Warsaw, 10 Romana Sanguszki Street, Mazowieckie",  
  "F_Country": "PL",  
  "F_Type": 3,  
  "F_Type_Other": "string",  
  "F_Status": true,  
  "F_ExciseNumber1": true,  
  "F_ExciseNumber2": "PL123456789AA",  
  "OtherFID_R": true,  
  "OtherFID_N": [  
    "ExampleFacility001"  
  ],  
  "Reg_3RD": true,  
  "Reg_EOID": "PLPWPW0000001"  
}
```

Success Response

```
HTTP/1.1 202 Accepted  
Connection: close  
Date: Tue, 05 Mar 2019 09:04:03 GMT  
Content-Type: application/json; charset=utf-8
```

```
{  
  "Request_Id": 681  
}
```

Deactivate facility



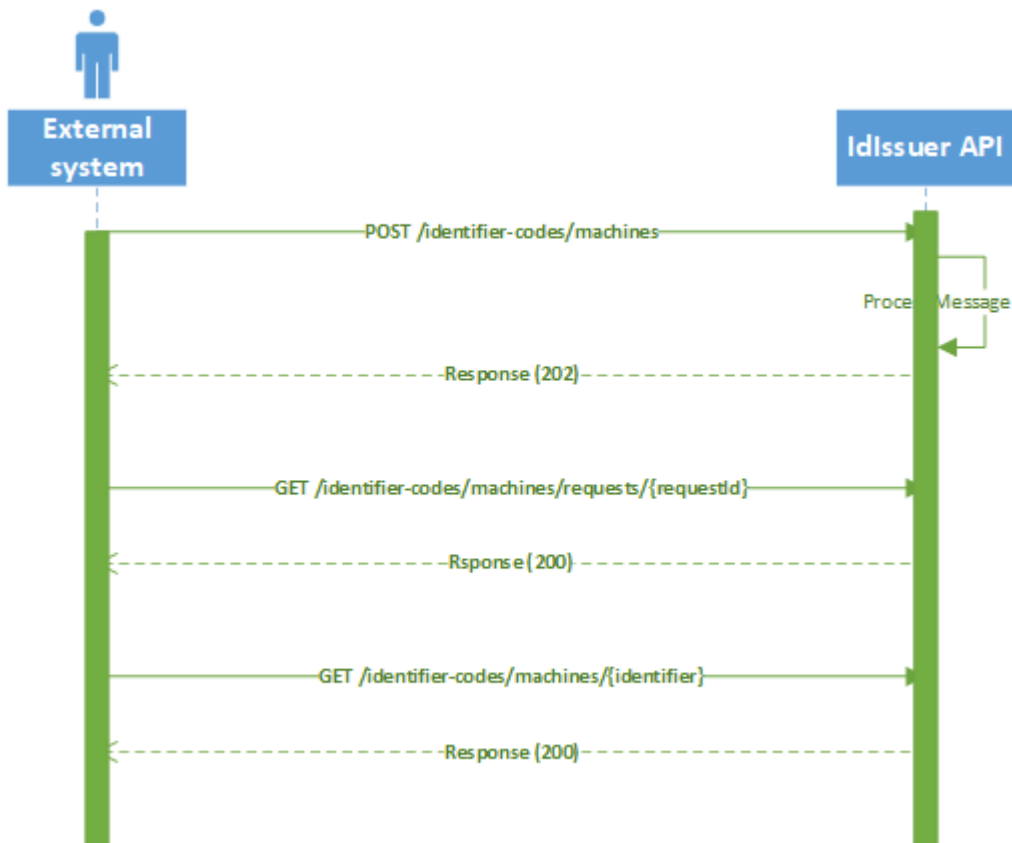
Data example

```
DELETE {{apiHost}}/identifier-codes/facilities/  
Content-Type: application/json  
Authorization: Bearer {{AUTH_TOKEN}}  
  
{  
  "EO_ID": "PLPWPWA1B210A",  
  "EO_CODE": "Ex4Mp13",  
  "F_ID": "PLPWPW987ABC",  
  "Reg_3RD": true,  
  "Reg_EOID": "PLPWPW0000001"  
}
```

Success Response

```
HTTP/1.1 202 Accepted  
Connection: close  
Date: Tue, 05 Mar 2019 09:04:03 GMT  
Content-Type: application/json; charset=utf-8  
  
{  
  "Request_Id": 681  
}
```

Register machine



Data example

```
POST {{apiHost}}/identifier-codes/machines/  
Content-Type: application/json  
Authorization: Bearer {{AUTH_TOKEN}}
```

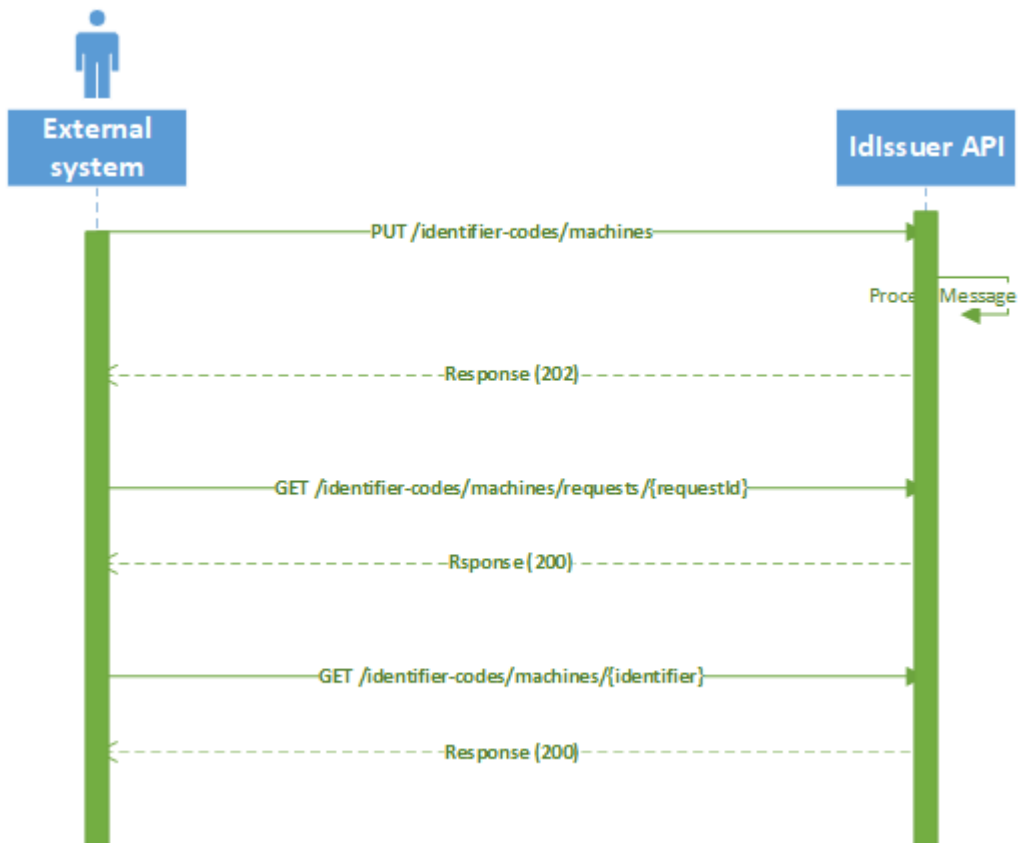
```
{  
  "EO_ID": "PLPWPWA1B210A",  
  "EO_CODE": "Ex4Mp13",  
  "F_ID": "PLPWPW987ABC",  
  "M_Producer": "Machina Maker M&M",  
  "M_Model": "Best Machine",  
  "M_Number": "BM-MM-12345",  
  "M_Capacity": 100000  
}
```

Success Response

```
HTTP/1.1 202 Accepted  
Connection: close  
Date: Tue, 05 Mar 2019 09:04:03 GMT  
Content-Type: application/json; charset=utf-8
```

```
{  
  "Request_Id": 681  
}
```

Update machine



Data example

```
PUT {{apiHost}}/identifier-codes/machines/  
Content-Type: application/json  
Authorization: Bearer {{AUTH_TOKEN}}
```

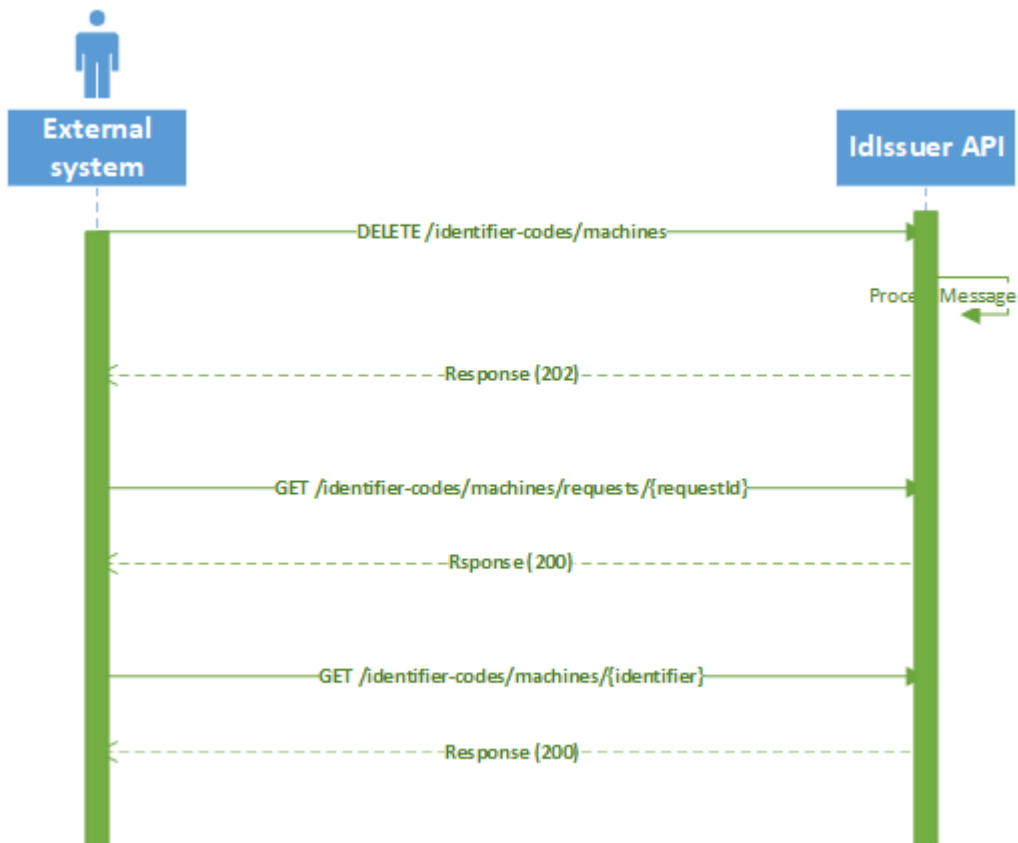
```
{  
  "M_ID": "PLPWPW00012ABC",  
  "EO_ID": "PLPWPWA1B210A",  
  "EO_CODE": "Ex4Mp13",  
  "F_ID": "PLPWPW987ABC",  
  "M_Producer": "Machina Maker M&M",  
  "M_Model": "Best Machine",  
  "M_Number": "BM-MM-12345",  
  "M_Capacity": 100000  
}
```

Success Response

```
HTTP/1.1 202 Accepted  
Connection: close  
Date: Tue, 05 Mar 2019 09:04:03 GMT  
Content-Type: application/json; charset=utf-8
```

```
{  
  "Request_Id": 681  
}
```

Deactivate machine



Data example

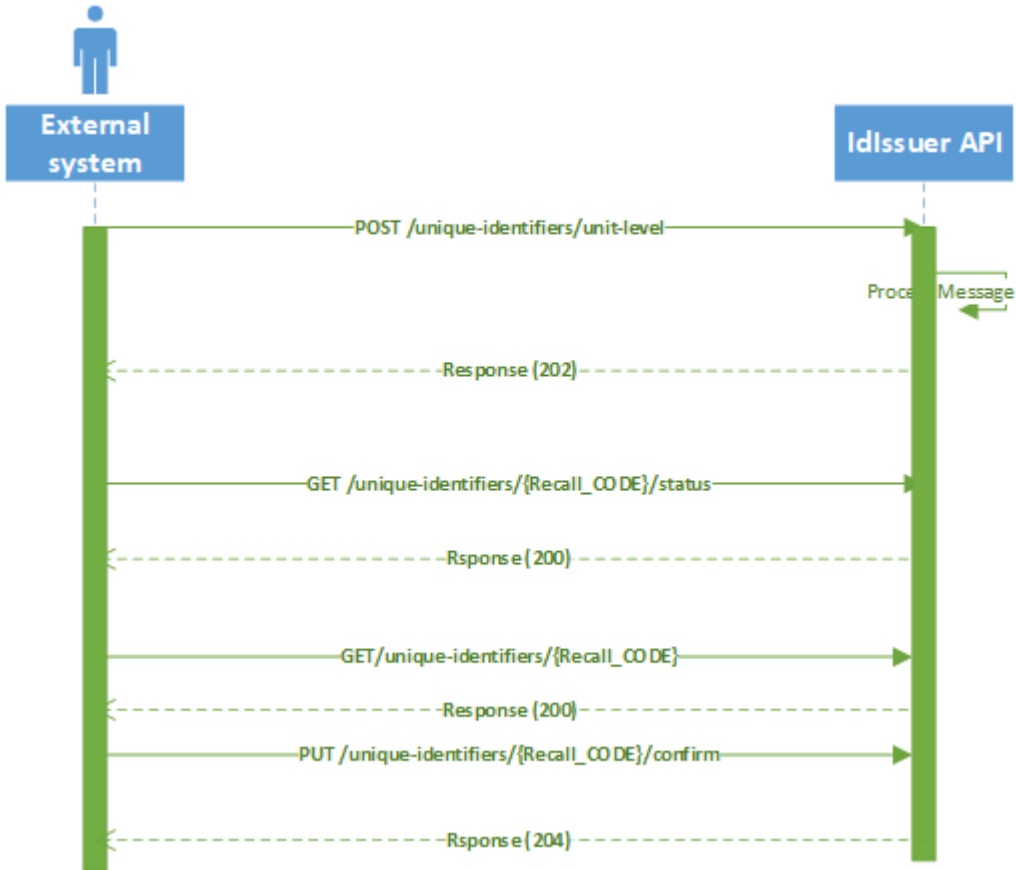
```
DELETE {{apiHost}}/identifier-codes/machines/  
Content-Type: application/json  
Authorization: Bearer {{AUTH_TOKEN}}  
  
{  
  "EO_ID": "PLPWPWA1B210A",  
  "EO_CODE": "Ex4Mp13",  
  "F_ID": "PLPWPW987ABC",  
  "M_ID": "PLPWPW00012ABC"  
}
```

Success Response

```
HTTP/1.1 202 Accepted  
Connection: close  
Date: Tue, 05 Mar 2019 09:04:03 GMT  
Content-Type: application/json; charset=utf-8  
  
{  
  "Request_Id": 681  
}
```

Unique identifiers requests flow

Request for unit level UIs



Data example

```
POST {{apiHost}}/unique-identifiers/unit-level
Content-Type: application/json
Authorization: Bearer {{AUTH_TOKEN}}

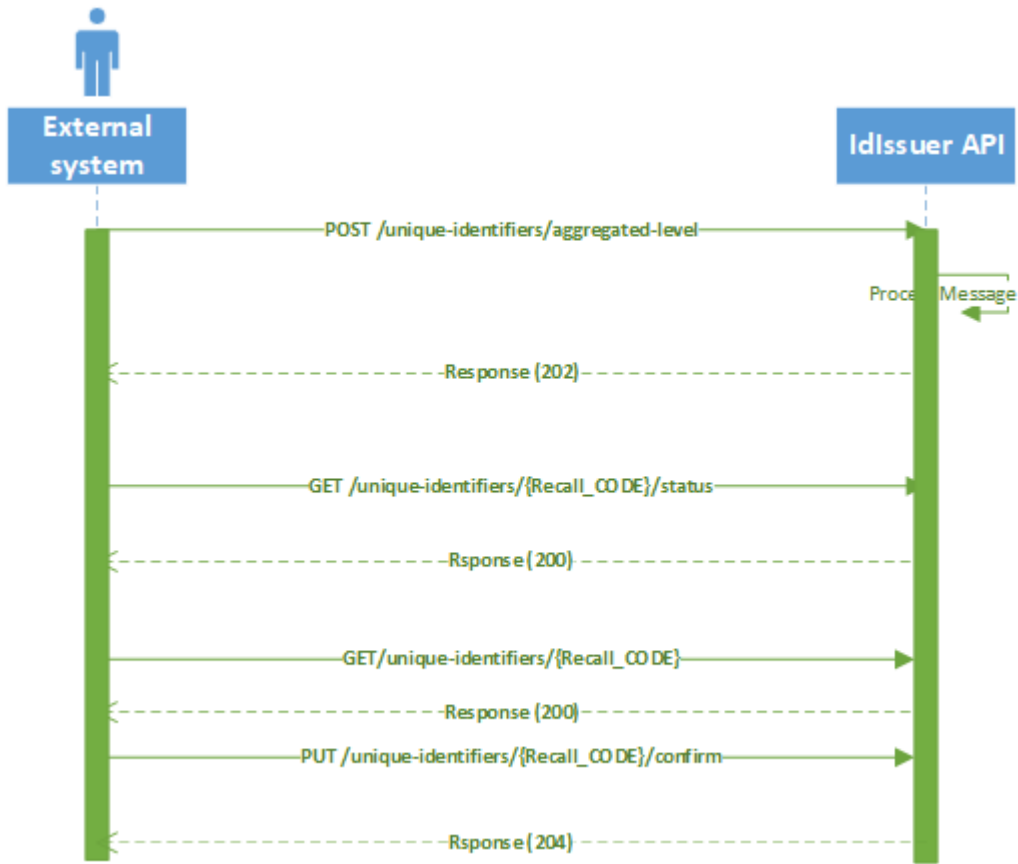
{
  "EO_ID": "PLPWPWA1B210A",
  "F_ID": "PLPWPW987ABC",
  "Process_Type": true,
  "M_ID": "PLPWPW00012ABC",
  "P_Type": 11,
  "P_OtherType": "Description of another type of tobacco product",
  "P_CN": "WWW1234EE",
  "P_Brand": "Special Cigaretts",
  "P_weight": 50,
  "TP_ID": "12345-22-54321",
  "TP_PN": "12345678",
  "Intended_Market": "F1",
  "Intended_Route1": true,
  "Intended_Route2": "AD",
  "Import": true,
  "Req_Quantity": 100000,
  "Process_Immediately": true
}
```


Success Response

```
HTTP/1.1 202 Accepted
Connection: close
Date: Tue, 05 Mar 2019 09:31:49 GMT
Content-Type: application/json; charset=utf-8
Location: {{apiHost}}/unique-identifiers/521
```

```
{
  "recall_code": 521
}
```

Request for aggregated level UIs



Data example

```

POST {{apiHost}}/unique-identifiers/aggregated-level
Content-Type: application/json
Authorization: Bearer {{AUTH_TOKEN}}

{
  "EO_ID": "PLPWPWA1B210A",
  "F_ID": "PLPWPW987ABC",
  "Req_Quantity": "100000",
  "Process_Immediately": true
}
  
```

Success Response

```

HTTP/1.1 202 Accepted
Connection: close
Date: Tue, 05 Mar 2019 09:31:49 GMT
Content-Type: application/json; charset=utf-8
Location: {{apiHost}}/unique-identifiers/521

{
  "recall_code": 521
}
  
```