



NATIONAL ORGANISATION FOR MEDICINES

Information Technology Division

GRELIS API

Application Developer's Guide

version 1.1

NATIONAL ORGANISATION FOR MEDICINES

GRELIS API Application Developer's Guide (v1.1)

Document version notes

1.0	15/05/2023	Initial
1.1	08/06/2023	Change in swagger-ui URL.

©EOF – National Organisation for Medicines
284 Mesogeion Av.
Holargos, 15562
Greece

Phone +30 213 2040 000 • Fax +30 210 65 45 535 • Email rest-services@eof.gr

Table of Contents

General REST principles	1
Service base URL.....	1
Security.....	1
Methods and return codes.....	4
Error response body	5
Paths operations.....	6
GET /attachments/verify.....	6
Schemas.....	7
Attachment.....	7
OpenAPI	7

General REST principles

REST API provides a powerful, convenient, and simple Web services API for interacting with the national shortage report database. Its advantages include ease of integration and development, and it's an excellent choice of technology for use with mobile applications and Web 2.0 projects.

Service base URL

The GreLIS API is provided in two different environments “Training” and “Production”. For your first steps please use the “Training” environment and also if you later test any changed functionality.

Use the “Production” environment only for production use cases!

To access the API in the different environments, use the following Links:

Environment	Endpoint
Training	https://services.eof.gr/grelis-api-uat/v1/
Production	https://services.eof.gr/grelis-api/v1/

Security

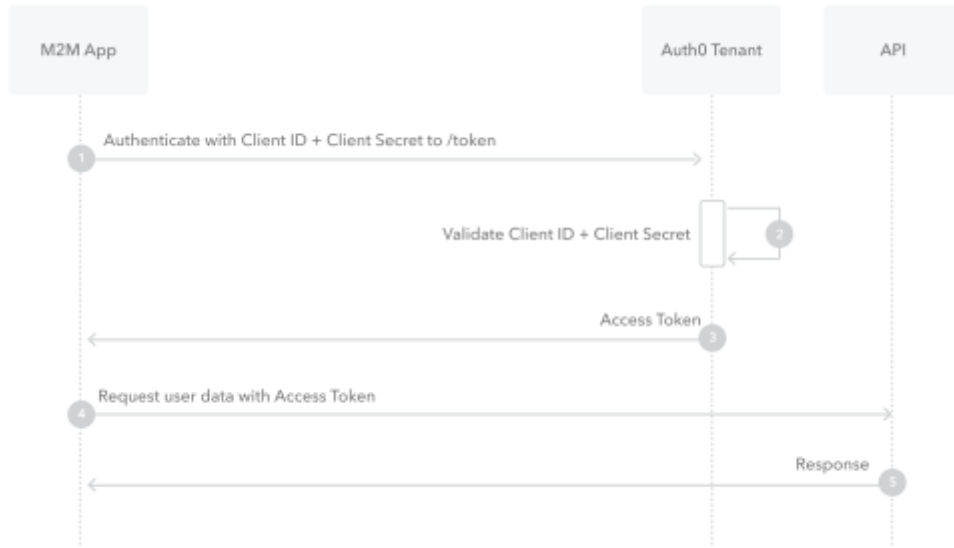
All REST-resources require a valid user/client to be authenticated. OAuth2, Client Credentials flow is used to acquire an authorization token. This token must be included in a HTTP-header for all request:

Authorization: Bearer <YourAuthorizationToken>

To get your <YourAuthorizationToken> use the following OAuth Endpoints for your authentication request:

Environment	Endpoint
Training	https://services.eof.gr/iam-uat/oauth2/token
Production	https://services.eof.gr/iam/oauth2/token

The Client Credentials Flow (defined in OAuth 2.0 RFC 6749, section 4.4) involves an application exchanging its application credentials, such as client ID and client secret, for an access token.



1. Application sends application's credentials to the Auth0 Authorization Server.
2. Auth0 Authorization Server validates application's credentials.
3. Auth0 Authorization Server responds with an access token.
4. Application can use the access token to call an API on behalf of itself.
5. API responds with requested data.

This flow requires the registration of a client. You can register a client at:

Environment	Endpoint
Training	https://services.eof.gr/iam-ext-uat
Production	https://services.eof.gr/iam-ext

On successful completion of a client registration, a client id and client secret shall be generated. To acquire an access token the following procedure is required:

Do a POST to the OAuth Endpoint:

https://services.eof.gr/iam/oauth2/token?grant_type=client_credentials&client_id=<your_client_id>&client_secret=<your_client_secret>&scope=label.attachment.verify

Methods and return codes

The following table displays HTTP-methods and corresponding operations.

Method	Operations
GET	Get a single resource or get a collection of resources.
POST	Create a new resource. Also used for executing resource-queries which have a too complex request-structure to fit in the query-URL of a GET-request.
PUT	Update properties of an existing resource. Also used for invoking actions on an existing resource.
DELETE	Delete an existing resource.

The following table displays HTTP-methods response codes.

Method	Operations
200 - Ok	The operation was successful and a response has been returned (GET and PUT requests).
201 - Created	The operation was successful and the entity has been created and is returned in the response-body (POST request).
204 - No content	The operation was successful and entity has been deleted and therefore there is no response-body returned (DELETE request).
400 - Bad Request	The 400 (Bad Request) status code indicates that the server cannot or will not process the request due to something that is perceived to be a client error.
401 - Unauthorized	The operation failed. The operation requires an Authentication header to be set. If this was present in the request, the supplied credentials are not valid or the user is not authorized to perform this operation.
403 - Forbidden	The operation is forbidden and should not be re-attempted. This does not imply an issue with authentication not authorization, it's an operation that is not allowed.
404 - Not found	The operation failed. The requested resource was not found.
405 - Method not allowed	The operation failed. The used method is not allowed for this resource.

415 - Unsupported Media Type	The operation failed. The request body contains an unsupported media type. Also occurs when the request-body JSON contains an unknown attribute or value that doesn't have the right format/type to be accepted.
500 - Internal server error	The operation failed. An unexpected exception occurred while executing the operation. The response-body contains details about the error.

The media-type of the HTTP-responses are **application/json** and **application/xml** unless binary content is requested, the media-type of the content is used.

The default media-type is **application/json**. You can request a response in XML by adding the header "Accept: application/xml".

Error response body

When an error occurs the response body contains an object describing the error that occurred:

```
{
  "timestamp": "2023-05-18T09:53:13.164+00:00",
  "status": 400,
  "error": "Bad Request",
  "message": "Required parameter 'stickerBarcode' is not present.",
  "path": "/grelis-api/v1/attachments/verify"
}
```


Paths operations

Parameters that are part of the url need to be properly escaped in case the segment contains special characters. Most frameworks have this functionality built in, but it should be taken into account. Especially for segments that can contain forward-slashes this is required.

GET /attachments/verify				
Description				
Returns attachment data from the system based on your search criteria				
Parameters				
Name	Located in	Description	Required	Example
productBarcode	Request parameter	The barcode of the medicinal product package. All barcodes start with 280.	Yes	/attachments/verify?productBarcode=2802765402017&stickerBarcode=220428191105
stickerBarcode	Request parameter	The barcode of the authenticity sticker	Yes	/attachments/verify?productBarcode=2802765402017&stickerBarcode=220428191105
Responses				
Code	Description	Example		
200	Attachment response	<pre>{ "productBarcode": "2802106503021", "productDescription": "HAGEVIR CREAM 5% W/W TUBX10G", "stickerBarcode": "070023968601", "lot": "05", "lotExpires": "201107" }</pre>		

Schemas

Attachment

Name	Description	Schema
productBarcode	The barcode of the medicinal product package. All barcodes start with 280.	string
productDescription	The description of the medicinal product package.	string
stickerBarcode	The barcode of the authenticity sticker.	string
lot	The lot number	string
lotExpires	The expiry date of the lot. The format is "YYYYMM"	string

OpenAPI

The GreLIS API description follows the OpenAPI Specification 3.

OpenAPI is a standard, language-agnostic interface to RESTful APIs which allows both humans and computers to discover and understand the capabilities of the service without access to source code, documentation, or through network traffic inspection. You can use this document to generate the client code.

The OpenAPI document (json format) and the UI for live testing are available at:

<https://services.eof.gr/grelis-api-uat/swagger-ui.html>

For more information visit the <https://swagger.io>