

# **Technical specification of the API software "Data Transfer Administrator Platform"**

**Version 1.1**

**02.08.2024**

# Table of contents

- 1. Getting an authorization token ..... 3
  - 1.1 The body of the request..... 3
  - 1.2 Successful execution of the request ..... 3
  - 1.3 Errors ..... 3
  - 1.4 Implementation in the PHP language ..... 3
- 2. Transfer signed XML file of the contract ..... 4
  - 2.1 The body of the request..... 4
  - 2.2 Successful execution of the request ..... 4
  - 2.3 Errors ..... 4
  - 2.4 Implementation in the PHP language ..... 5
- 3. Receiving the status of processing the sent XML file of the contract with an attached signature in base64 format..... 5
  - 3.1 The body of the request..... 6
  - 3.2 Successful execution of the request ..... 6
  - 3.3 Errors ..... 6
  - 3.4 Implementation in the PHP language ..... 6
- 4. Get information about all downloaded signed XML files for the selected date ..... 7
  - 4.1 The body of the request..... 7
  - 4.2 Successful execution of the request ..... 7
  - 4.3 Errors ..... 7
  - 4.4 Implementation in the PHP language ..... 7

## 1. Getting an authorization token

To access the API functionality, you need to receive an authorization token that will be required to access and upload data. In this POST request, it is necessary to include a signed file in base64 format as a parameter, the signature is imposed by the person who has registered on the platform. The link where the POST request is made with the corresponding parameter:

<https://apd.oree.com.ua/api/tokens/create>

<https://apd-test.oree.com.ua/api/tokens/create>

### 1.1 The body of the request

| Parameters | Description  |
|------------|--|
| file       | a parameter containing a signed XML file in base64 format. |

An example of a signed XML file for receiving an authorization token:

```
<?xml version="1.0" encoding="utf-8"?>
<AuthRequest>
  <edr pou>EDRPOU_CODE</edr pou>
  <drfo>VAT_CODE</drfo>
</AuthRequest>
```

### 1.2 Successful execution of the request

If the signature file has been validated and the user has entered the correct data in the signed XML file available in the system, the request will return the following response in JSON format with status 201:

```
{"status_code": "0", "message": "Authorization token created", "token": "YOUR_AUTH_TOKEN"}.
```

### 1.3 Errors

A list of errors that the user may receive.

| The code that returned the request | Description  |
|------------------------------------|--|
| 404                                | there is no signed XML file in base64 format, the server returns a response in JSON format.  |
| 400                                | the error is related to a signed XML file, the file is corrupted or it is impossible to obtain signed data, the response is provided in JSON format. |
| 408                                | the signature time and the current time have a difference of more than 30 seconds, the server returns a response in JSON format.                     |
| 401                                | the user is not found according to the data entered in the XML file, the server returns a response in JSON format.                                   |

An example of a JSON response from the server when an error is received:

```
{
  "status_code": "-10001",
  "message": "Sended data incorrect!"
}
```

### 1.4 Implementation in the PHP language

```
<?php
$curl = curl_init();
curl_setopt_array($curl, array(
  CURLOPT_URL => 'https://apd.oree.com.ua/api/tokens/create',
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_ENCODING => "",
  CURLOPT_MAXREDIRS => 10,
```

```

CURLOPT_TIMEOUT => 0,
CURLOPT_FOLLOWLOCATION => true,
CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
CURLOPT_CUSTOMREQUEST => 'POST',
CURLOPT_POSTFIELDS => array('file' => 'YOUR_SIGNED_XML_ENCODED_IN_BASE64'),
));
$response = curl_exec($curl);
curl_close($curl);
echo $response;
?>

```

## 2. Transfer signed XML file of the contract

The POST request contains 2 parameters, namely : file - a signed file in bas64 format and type - a parameter indicating the type of contract, taking one of the values "nonstandard" or "standard". The request header must contain the Authorization parameter, which accepts the value of the token received in clause 1 of this instruction. The value of the Authorization parameter has the following format: Bearer YOUR\_AUTH\_TOKEN (an example of the request is posted in clause 2.4 of this instruction). If the request is successfully executed, the user will receive a request status of 200 and the corresponding id number, by which you will be able to track the status of the uploaded file verification. Processing of files from users is carried out in the order of receipt and may take some time. The processing status is obtained by request as described in clause 3 of this manual.

The XML file must be generated in accordance with the XSD schema available at the following link: [non-standard contract](#) and [standard contract](#).

The time between signing and downloading the signed XML file **should not exceed 30 seconds**.

The link through which the POST request is made with the appropriate parameters:

<https://apd.oree.com.ua/api/tokens/createFile>

<https://apd-test.oree.com.ua/api/tokens/createFile>

### 2.1 The body of the request

| Parameters | Description  |
|------------|--|
| file       | a parameter containing a signed file of a standard or non-standard contract in base64 format.            |
| type       | a parameter containing the value "nonstandard" or "standard", which corresponds to the type of contract. |

### 2.2 Successful execution of the request

If the signature file has been verified and the user has entered the correct data in the signed XML file, the request will return the following response in JSON format with a status of 200.

```

{"id":12,"status_code":"0","message":"Data has been successfully uploaded"}

```

### 2.3 Errors

A list of errors that the user may receive.

| The code that returned the request | Description  |
|------------------------------------|--|
| 404                                | signed contract file not found, no XML in base64 format, the server returns a response in JSON format. |
| 406                                | the contract type parameter is not found, the server returns a response in JSON format.                |

|     |  |
|-----|--|
| 417 | the contract type parameter is incorrect, the value should be "nonstandard" or "standard", the server returns a response in JSON format. |
| 400 | an error occurred while verifying the signed file, the server returns a response in JSON format.   |
| 403 | the user who signed the file is not found in the system, the server returns a response in JSON format.                                   |
| 412 | the signed XML file does not match the structure of the XSD template, the server returns a response in JSON format.                      |

An example of a JSON response from the server when an error is received:

```
{
  "status_code": "-10004",
  "message": "Base64 signed xml file string not found!"
}
```

## 2.4 Implementation in the PHP language

```
<?php
$curl = curl_init();
curl_setopt_array($curl, array(
  CURLOPT_URL => 'https:// apd.oree.com.ua/api/tokens/createFile',
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_ENCODING => "",
  CURLOPT_MAXREDIRS => 10,
  CURLOPT_TIMEOUT => 0,
  CURLOPT_FOLLOWLOCATION => true,
  CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
  CURLOPT_CUSTOMREQUEST => 'POST',
  CURLOPT_POSTFIELDS => array('file' => 'YOUR_SIGNED_XML_ENCODED_IN_BASE64','type' => 'nonstandard'),
  CURLOPT_HTTPHEADER => array(
    'Authorization: Bearer YOUR_AUTH_TOKEN '
  ),
));
$response = curl_exec($curl);
curl_close($curl);
echo $response;
?>
```

## 3. Receiving the status of processing the sent XML file of the contract with an attached signature in base64 format

POST-request containing id parameter, which was received when uploading signed contract file in base64 format (clause 2 of this instruction). The request header must contain the Authorization parameter, which accepts the value of the token received in clause 1 of this instruction. The value of the Authorization parameter has the following format: Bearer YOUR\_AUTH\_TOKEN (an example of a request is posted in clause 2.4 of this instruction). The link through which the POST request is made with the appropriate parameters:

[https://apd.oree.com.ua/api/tokens/checkStatus?id=your\\_id\\_here](https://apd.oree.com.ua/api/tokens/checkStatus?id=your_id_here)

[https://apd-test.oree.com.ua/api/tokens/checkStatus?id=your\\_id\\_here](https://apd-test.oree.com.ua/api/tokens/checkStatus?id=your_id_here)

### 3.1 The body of the request

| Parameters | Description   |
|------------|---|
| id         | a parameter contained in the GET request URL, a numeric type. |

### 3.2 Successful execution of the request

If the user is successfully authorized with a token and the id parameter of the uploaded and signed XML file is available, the following response will be received in JSON format:

```
{"status_code": "0", "message": "OK", "status": "1", "error": ""}
```

Response statuses in the request:

- 1 - Queued for processing
- 2 - Error in the XML file
- 3 - File processing is successful

If the status is "2", the error description will be contained in the "error" parameter.

```
{"status_code": "2", "message": "OK", "status": "2", "error": "Помилка валідації XML. Текст помилки: Element '{http://www.oree.com.ua/REMIT/MONonStandardContract_V1.xsd} tradingCapacity': [facet 'enumeration'] The value 'P11' is not an element of the set {'P', 'A'}."}
```

### 3.3 Errors

List of errors that the user may receive

| The code that returned the request | Description   |
|------------------------------------|---|
| 404                                | data for the selected id is not found, the server returns a response in JSON format.                            |
| 424                                | the parameter passed by GET id is empty or has an invalid format, the server returns a response in JSON format. |

An example of a JSON response from the server when an error is received:

```
{  
  "status_code": "-10009",  
  "message": "No data found"  
}
```

### 3.4 Implementation in the PHP language

```
<?php  
$curl = curl_init();  
curl_setopt_array($curl, array(  
  CURLOPT_URL => 'http://apd.oree.com.ua/api/tokens/checkStatus?id=338',  
  CURLOPT_RETURNTRANSFER => true,  
  CURLOPT_ENCODING => "",  
  CURLOPT_MAXREDIRS => 10,  
  CURLOPT_TIMEOUT => 0,  
  CURLOPT_FOLLOWLOCATION => true,  
  CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,  
  CURLOPT_CUSTOMREQUEST => 'POST',  
  CURLOPT_HTTPHEADER => array(  
    'Authorization: Bearer YOUR_AUTH_TOKEN'  
  ),  
));  
$response = curl_exec($curl);  
curl_close($curl);  
echo $response;  
?>
```

## 4. Get information about all downloaded signed XML files for the selected date

POST request containing one date parameter, which is necessary to obtain information about all uploaded files for the selected date. The request header must contain the Authorization parameter, which accepts the value of the token received in paragraph 1 of this instruction. The value of the Authorization parameter has the following format: Bearer YOUR\_AUTH\_TOKEN (an example of a request is posted in clause 2.4 of this instruction). The link by which the POST request with the corresponding parameter is made:

<https://apd.oree.com.ua/api/tokens/getAllUploadsByDay?date=09.07.2024>

<https://apd-test.oree.com.ua/api/tokens/getAllUploadsByDay?date=09.07.2024>

### 4.1 The body of the request

| Parameters | Description  |
|------------|--|
| date       | a parameter contained in the GET request URL in the format dd.mm.yyyy. |

### 4.2 Successful execution of the request

If the user is successfully authorized with a token and the uploaded data for the selected date is available, the following response will be received in JSON format:

|   |
|---|
| <pre>{   "status_code": "0",   "data": "[     {\"id\": \"351\", \"created_at\": \"2024-07-17 14:50:06\"},     {\"id\": \"352\", \"created_at\": \"2024-07-17 14:50:21\"},     {\"id\": \"342\", \"created_at\": \"2024-07-17 14:01:18\"},     {\"id\": \"343\", \"created_at\": \"2024-07-17 14:01:23\"},     {\"id\": \"344\", \"created_at\": \"2024-07-17 14:29:09\"},     {\"id\": \"349\", \"created_at\": \"2024-07-17 14:36:40\"},     {\"id\": \"350\", \"created_at\": \"2024-07-17 14:37:18\"},     {\"id\": \"345\", \"created_at\": \"2024-07-17 14:31:23\"},     {\"id\": \"346\", \"created_at\": \"2024-07-17 14:32:29\"},     {\"id\": \"347\", \"created_at\": \"2024-07-17 14:34:23\"},     {\"id\": \"348\", \"created_at\": \"2024-07-17 14:36:03\"}   ]",   "message": "List uploaded files by date" }</pre> |
|---|

### 4.3 Errors

A list of errors that the user may receive.

| The code that returned the request | Description  |
|------------------------------------|--|
| 400                                | the format of the transmitted date is incorrect, the server returns a response in JSON format. |

An example of a JSON response from the server when an error is received:

|   |
|---|
| <pre>{   "status_code": "-10012 ",   "message": " Wrong date format. Date must be in format d.m.Y!" }</pre> |
|---|

### 4.4 Implementation in the PHP language

|   |
|---|
| <pre>&lt;?php \$curl = curl_init();</pre> |
|---|

```
curl_setopt_array($curl, array(
  CURLOPT_URL => 'http:// apd.oree.com.ua/api/tokens/getAllUploadsByDay?date=09.07.2024',
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_ENCODING => "",
  CURLOPT_MAXREDIRS => 10,
  CURLOPT_TIMEOUT => 0,
  CURLOPT_FOLLOWLOCATION => true,
  CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
  CURLOPT_CUSTOMREQUEST => 'GET',
  CURLOPT_HTTPHEADER => array(
    'Authorization: Bearer YOUR_AUTH_TOKEN'
  ),
));
$response = curl_exec($curl);
curl_close($curl);
echo $response;
?>
```