



TEXT FINDER API

Developers Guide

[Content description](#)

This is a reference manual and configuration guide for the NeoCheck Text Finder API product. It shows how to interact with the JSON Web API from an external client to search texts in a given document and the degree of correlation between the search text and the text found.

Madrid, June 27th 2019

Disclaimer

The information contained within this document is and shall remain the property of NeoCheck. It must not be produced in whole or in part, or given or communicated to any third party without the prior consent of NeoCheck.

Whilst careful attention has been paid to ensure the appropriate referencing of information and sources, it is possible that a document of this nature might contain some errors. In such cases, upon notification NeoCheck will immediately analyze and make any required corrections or amendments.

©NeoCheck 2019



Release Notes

Version	Date	Description
1.0	27/06/2019	Initial Version



Index

Release Notes	2
1. NeoCheck API Endpoint	4
2. Authentication	5
2.1. Obtain JWT Authentication Token	5
2.2. Token Refresh	6
2.3. Abusive usage countermeasures	7
3. Text Finder Check.....	8



1. NeoCheck API Endpoint

NeoCheck provides an API for address verification checks. This API allows users to check if a given address exists and correlate it against a given document, such as bills, contracts, etc.

Endpoint	Environment
https://neocheck.net/api	PRO



2. Authentication

The Address Check API access is protected by using basic authentication and OAuth 2.0 authorization tokens.

2.1. Obtain JWT Authentication Token

HTTPS POST method to authenticate. In order to request access to Address Check API, you must call the authentication method with your API `client_id` and `client_secret`, along with a valid `username` and `password`. Once validated, you will receive an authorization token, which must be used to call Address Check API methods.

Endpoint	HTTP Method
<code>connect/token</code>	POST

Request Parameters:

Name	Type	Description
<code>content_type</code>	string	<code>application/x-www-form-urlencoded</code> indicates that parameters are passing as key/value pairs in the body of the HTTPS message
<code>grant_type</code>	string	<code>password</code> indicates that it's an authentication request including basic authentication
<code>client_id</code>	string	your Client Id
<code>client_secret</code>	string	Your client secret
<code>username</code>	string	valid username already created in the system for the Client Id
<code>password</code>	string	valid password for username provided

Response:

If request is correct, you will receive a Success response with the access token generated. This token is only valid for a short period, exposed in the `expires_in` property. You will also receive a `refresh_token`, which allows you to refresh your access token without having to send your credentials again. The API call to refresh token is explained in the next point.

HTTP Code	Body	Description
200	JSON Authorization object	Valid OAuth token for request authorization
400	JSON ErrorMessage object	Invalid ClientId
400	JSON ErrorMessage object	Unsupported grant type
400	JSON ErrorMessage object	Invalid credentials
500	JSON ErrorMessage object	Internal Error



Example of Successful Authorization Response:

```

ResultCode: HTTP/1.1 200 OK
Content-Type: application/json
Body:
{
  "token_type": "Bearer",
  "access_token": "yourAccessToken",
  "expires_in": 3600,
  "id_token": "yourIdToken",
  "refresh_token": "yourRefreshToken"
}

```

Example of Invalid Client Response:

```

ResultCode: HTTP/1.1 400 BadRequest
Content-Type: application/json
Body:
{
  "error": "Invalid ClientId",
}

```

2.2. Token Refresh

If you already have authenticated, you can use the `refresh_token` property provided to renew your `access_token` authorization token, which must be used to call Address Check API methods.

Endpoint	HTTP Method
<code>connect/token</code>	POST

Request Parameters:

Name	Type	Description
<code>content_type</code>	string	<code>application/x-www-form-urlencoded</code> indicates that parameters are passing as key/value pairs in the body of the HTTPS message
<code>grant_type</code>	string	<code>refresh_token</code> indicates that it's an authentication request including basic authentication
<code>refresh_token</code>	string	the refresh token
<code>client_id</code>	string	your Client Id
<code>client_secret</code>	string	Your client secret



Response:

If request is correct, you will receive a Success response (StatusCode 200), with a new access token, and the same refresh_token. If request is not correct, you will receive a Bad Request response (StatusCode 400), with error description

HTTP Code	Body	Description
200	JSON Authorization object	Valid OAuth token for request authorization
400	JSON ErrorMessage object	Invalid ClientId
400	JSON ErrorMessage object	Unsupported grant type
400	JSON ErrorMessage object	Invalid ticket (refresh_token not valid)
500	JSON ErrorMessage object	Internal Error

2.3. Abusive usage countermeasures

In order to prevent a bad or abusive usage and assure the best performance, there is a maximum number of requests per hour that a user can perform. This maximum number is set to 3600 calls per hour (an average of 1 call per second). After this limit is reached, the Address Check API will respond with a BadRequest (StatusCode 400) and an error message stating that the maximum number of calls per hour has been reached.



3. Text Finder Check

HTTPS POST method to check the existence of some data in a given document

Endpoint	HTTP Method
/v1/TextFinder	POST

Request Parameters:

Name	Type	Description
authorization	query parameter	Bearer access_token authorization header with a valid access token
textFinderRequest	Body: JSON object	JSON object, with properties: <ul style="list-style-type: none"> - searchTexts: list of data to search in a given document. Each searchText has properties Name and Value - documentImagesBase64: collection of document page images in Base64String format where the data to search is supposed to be present

Response:

If request is correct, you will receive a Success response along with an object containing a list of results including original request data and a list of results found in the document provided.

HTTP Code	Body	Description
200	List<SearchTextResult>	- List<SearchTextResults> : a list of results found in the document provided. Each result includes the requested SearchText, along with a list of hits found in the given document, with information about the page index, coordinates x and y, extracted text and accuracy respect to the request text
400		Invalid content
401		Unauthorized. Invalid token
500	JSON ErrorMessage object	Internal Error



Example of Successful Check Response:

```
ResultCode: HTTP/1.1 200 OK
Content-Type: application/json
Body:
[
  {
    "name": "ContractNumber",
    "value": "FG98787",
    "hits": [
      { "pageIndex": 0,
        "locationX": "25",
        "locationY": "94",
        "extractedText": "FG98787",
        "accuracy": 100},
      { "pageIndex": 1,
        "locationX": "78",
        "locationY": "225",
        "extractedText": "FG98787",
        "accuracy": 100}
    ]
  },
  {
    "name": "Address",
    "value": "445 Mount Eden Road, Mount Eden, Auckland",
    "hits": [
      { "pageIndex": 0,
        "locationX": "41",
        "locationY": "85",
        "extractedText": "445 Mount Eden Rd Mount Eden Auckland",
        "accuracy": 100},
      { "pageIndex": 0,
        "locationX": "66",
        "locationY": "58",
        "extractedText": "445 Mount Eden Rd",
        "accuracy": 80}
    ]
  }
]
```

