PlagAware Plagiarism Checker API V2.2

Last Update: 2024-10-10

Introduction

PlagAware is an award-winning professional plagiarism scan engine that can be integrated easily and seamlessly with any desktop, web or mobile apps. This allows your app to perform plagiarism scans without having your users visiting PlagAware or even being aware that PlagAware scans are being performed in the background.

Prerequisites

All API calls require you to have registered a valid PlagAware user account and sufficient ScanCredits available to request plagiarism scans. API calls require a user code parameter as part of their call signature. In order to obtain a user code, please visit https://www.plagaware.com/service/api and request your personal user code. Please keep this user code confidential as it allows access to all API calls on your behalf.

Getting Started

Please follow the getting started workflow to get your app up to speed in minutes:

1. Create your personal PlagAware user account at https://www.plagaware.com/register
2. Create your PlagAware UserCode (API Key) at https://www.plagaware.com/service/api
3. Implement code for submitting texts from your app to PlagAware using the *SubmitText* API call
4. (a) Implement server side callback script waiting to be called by PlagAware once plagiarism check is finalized (see example below) – or –
   (b) Poll for status changes of the requested report using the *GetReportMetadata* API call

Submit Text for Plagiarism Scan

- Description: Submit a text or a file document containing text to be scanned for plagiarism. This call corresponds to the manual submission of text documents via PlagAware's user front end (https://www.plagaware.com/plagiarism_detection/new-plagiarism-scan).
- Endpoint: https://www.plagaware.com/service/api
- Parameters:
  - UserCode: User code as provided via https://www.plagaware.com/service/api
  - TestText: URL encoded text to be checked for plagiarism. Minimum length 250 characters. Do not specify in case a test file is to be submitted.
  - TestFile: Document in any supported format (e.g. PDF) to be submitted for plagiarism check. If specified, this will override TestText.
  - ResultUrl: URL encoded callback URL of your web app. This URL will be called by PlagAware once the scan has been completed.
  - ReportName: Optional title of the plagiarism check. In case that this parameter is not specified, it will be generated from the submitted text.
  - ReportComment: Optional comment for the plagiarism check.
  - ProjectId: Associated id of the project which should be associated to the performed plagiarism scan. Project ids can be obtained via PlagAware's UI (https://www.plagaware.com/plagiarism_detection/projects)

- o DryRun: If the DryRun option is set, no plagiarism check will be requested. Instead, all parameters will be tested and the ResultUrl will be called with a random result value between 0% and 100%. This allows for testing of your integration.
- Return Value: Error code or JSON object
  - o Id: Report id, to be submitted to subsequent API calls, e.g. GetReportMetadata
  - o Words: Total number of words submitted as test text
  - o Status: Current status of the report, e.g. Scheduled
- Example:
  - o curl -k -X POST "https://www.plagaware.com/service/api" -d\
    "UserCode=385822e359afa26d52b5b286226f2cea\
    &ResultUrl=https%3A%2F%2Fmyplagtestapp.com%2Fcallback.php\
    &TestText=Plagiarism%20is%20not%20the%20same%20as%20copyright%20infringe
    ment.%20While%20both%20terms%20may%20apply%20to%20a%20particular%20a
    ct%2C%20they%20are%20different%20concepts%2C%20and%20false%20claims%20
    of%20authorship%20generally%20constitute%20plagiarism%20regardless%20of%20
    whether%20the%20material%20is%20protected%20by%20copyright"
- Remarks:
  - o Please visit https://www.plagaware.com/service/api for a working example of how to submit documents from a PHP script.

Get Report Metadata

- Description: Get all relevant metadata for a given report along with detailed information on identified sources.
- Endpoint: https://www.plagaware.com/api/
- Parameters:
  - o cmd=GetReportMetadata
  - o reportId=[numeric, id of report to be queried.]
  - o userCode=[alphanumeric, as provided via https://www.plagaware.com/service/api]
- Return Value: JSON object containing all relevant data regarding one specific plagiarism scan
  - o Name: Name of plagiarism scan report
  - o Lang: Language of report
  - o Status: Current status, e.g. ok, error, scheduled, active
  - o Credits: Total costs in ScanCredits
  - o TotalWords: Length of test text in words
  - o PlagWords: Length of plagiarized content in words
  - o ResultPercent: Percentage of plagiarized vs. total words
  - o HtmlLink: Static link to result report in HTML format
  - o PdfLink: Static link to result report in PDF format
  - o Author: Author of plagiarism scan text
  - o Comment: Provided comment of plagiarism scan text
  - o Scheduled: Scheduled time stamp
  - o Started: Started time stamp
  - o Completed: Completed time stamp
  - o Settings: Used settings to perform plagiarism scan
  - o Sources: Details on all identified sources
- Example:
  - o curl -X POST "https://www.plagaware.com/api/" -d \
    "cmd=GetReportMetadata \

&reportId=12345 \
&UserCode=yourUserCode"

- Remarks:
    - User identified by UserCode must match report owner

Example Callback Script (PHP)

The callback script might be implemented within your web app to prevent continuous polling for status changes of the submitted plagiarism scan. It will be called by PlagAware once the scan has been finalized.

// Assumption: Plagiarism check has been requested via API call (see above)

// fetch parameters from callback
```
$result = $_REQUEST['result'];
$id = $_REQUEST['id'];
$status = $_REQUEST['status'];
```

// Assemble link to pdf file from PlagAware
// As an alternative, you might want to invoke GetReportMetadata retrieving the PDF link
```
$pdfLink = "http://www.plagaware.com/reportpreview/?id=".$id."&pdf=1";
```

// copy pdf file to local file
```
$pdfFile = FILE_PATH."/report.pdf";
$contents = file_get_contents($pdfLink);
$result = file_put_contents($pdfFile, $contents);
```

// do whatever your app intends to do with the report.