
HUZAR API

Document version: 1.25

Date: 22.05.2023

TABLE OF CONTENTS

1.	Document information	4
2.	Changelog.....	4
3.	Terms used in this document.....	4
4.	HuzarAPI installation.....	5
3.1.	HuzarAPI configuration	5
3.2.	WinSAD Service configuration.....	7
3.3.	Running WinSAD as a service	9
3.4.	HuzarAPI installation test	9
5.	HuzarAPI with http protocol	11
4.1.	Methods using http GET	11
5.1.1.	GETSAD.....	11
5.1.2.	GETSADSTATUS	11
5.1.3.	COPYSAD	12
5.1.4.	GETSADBYRANGE	12
5.1.5.	GETATTACHEDDOCLIST	14
5.1.6.	GETATTACHEDDOCDET	15
5.1.7.	GETCUSTOMSMSGLIST.....	15
5.1.8.	GETGENERATEDDOCLIST.....	16
5.1.9.	GETREPRESENTATION	16
5.1.10.	GETDELIVERIES.....	17
5.1.11.	SENDSAD	17
4.2.	Methods using http POST.....	18
5.2.1.	ADDSAD.....	18
6.	Data formats	20
5.1.	Customs declaration list	20
5.2.	Documents attached to Customs declaration.....	20
7.	Examples	21
6.1.	GETSAD.....	21
7.1.1.	Declaration with a certain ID	21
7.1.2.	Error message	21

6.2.	GETSADBYRANGE	21
6.2.1	Date range.....	21
6.2.2	Headers	21
6.2.3	Last changes	22
6.2.4	Error message	22
6.3.	GETSADSTATUS	22
6.3.1.	Request on the SAD status.....	22
6.4.	GETCUSTOMSMSGLIST	23
6.4.1	Request on the SAD customs msg list.....	23
6.5.	GETGENERATEDDOCLIST	23
6.5.1	Request on the SAD generated docs	23
6.6.	GETREPRESENTATION	23
6.6.1	Request on the company's representations.....	23
6.7.	GETDELIVERIES	23
6.7.1	Request on MagCS deliveries.....	23
6.8	SENDSAD.....	24
6.8.1	Request on sending a declaration.....	24
6.9	ADDSAD	24
6.9.1	Request on adding a declaration	24

1. DOCUMENT INFORMATION

This document describes the functionality of the HuzarAPI module and data formats used for data exchange with Huzar Software systems. It also provides examples on how to use functions provided by the API.

2. CHANGELOG

Version	Change
1.24 2022-12-01	The version of this document was changed (previously 1.15) so it reflects the version of the file format (currently 1.24).
1.24 2022-12-01	New columns were added in specification - „Change version” and „Change description”. All changes from the latest versions are in red. Specification 1.24 is processed by WinSAD 18.00 and HuzarAPI 2.30.
1.24 2022-12-01	Added new fields for AIS4, AES+ and NCTS+ systems. AIS4 applies to all (apart H7) import declaration from 1.1.2023. AES+ (export) i NCTS+ (transit, NCTS phase 5) will apply to some export and transit documents in 2023, but the exact date is currently unknown. AES+ and NCTS+ will gradually replace current AES and NCTS systems.
1.25 2023-05-22	Added new fields for AES+ and NCTS+ systems. Specification 1.25 is processed by WinSAD 19.00 and HuzarAPI 2.40.

3. TERMS USED IN THIS DOCUMENT

Below you will find the list of terms used in this document:

- API – term used to describe the interface for communication with Huzar Software applications;
- HuzarAPI – name of the module (a plugin to Huzar Software applications) enhancing the standard functionality of the application with additional features;
- WinSAD – Huzar Software application designed to prepare customs documents and communicate with the customs systems of the Ministry of Finance.

4. HUZARAPI INSTALLATION

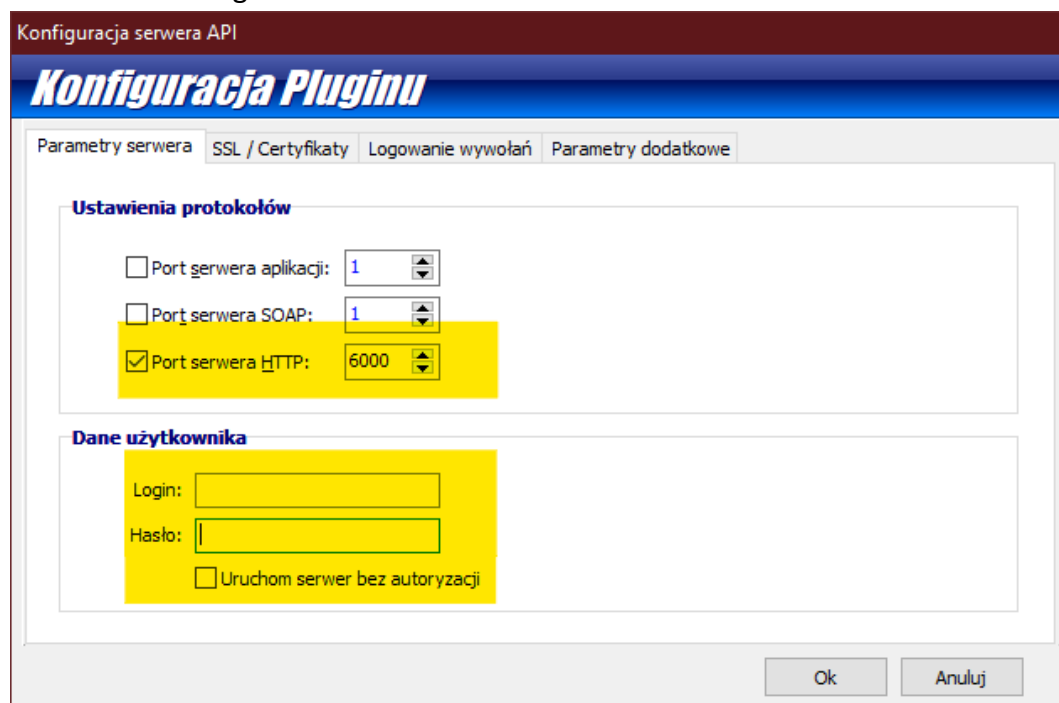
The HuzarAPI module is delivered as a DLL library (HSPlugAPI.dll file). This file should be placed in the WinSAD installation folder (i.e. the location of WinSADIB.exe file for Firebird database or WinSADMS.exe file for MSSQL Server database).

To start the module, you must perform the following steps:

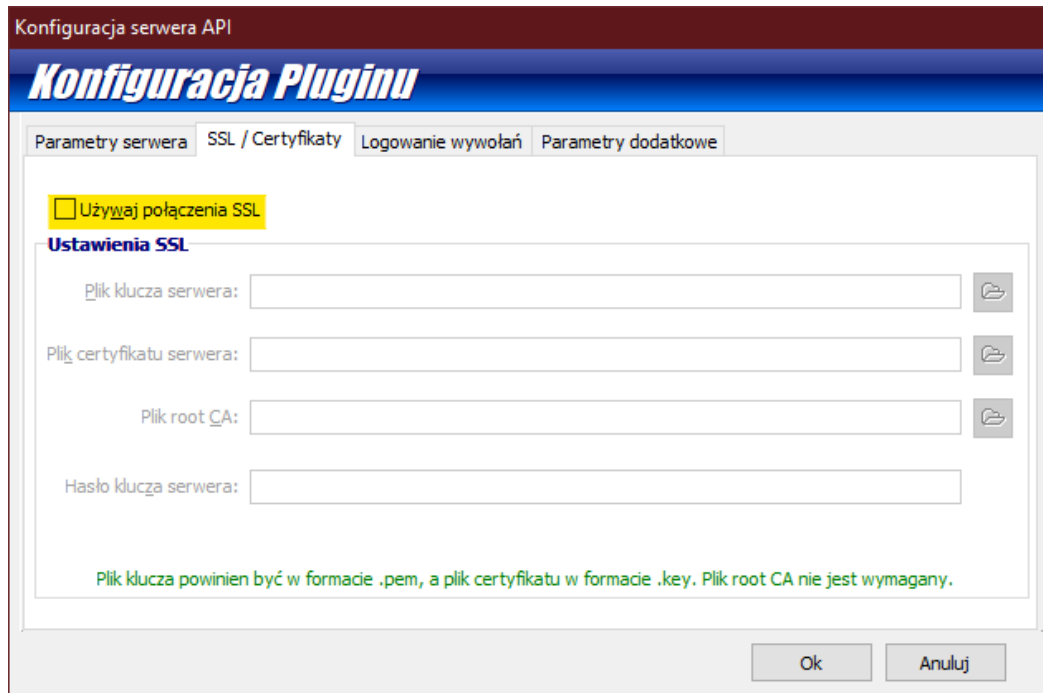
3.1. HUZARAPI CONFIGURATION

Start WinSAD and connect it to the database to which the API is to be connected.

After the first start of WinSAD, the configuration dialog of the HuzarAPI will be displayed, in which basic configuration data must be entered.



- *Port serwera HTTP (HTTP server port)* – API supports the HTTP protocol only. Enter the port number you want to be used by the API service.
- *Login i Hasło (Login and password)* – authorization data to access the API.



Konfiguracja serwera API

Konfiguracja Pluginu

Parametry serwera | **SSL / Certyfikaty** | Logowanie wywołań | Parametry dodatkowe

Używaj połączenia SSL

Ustawienia SSL

Plik klucza serwera:

Plik certyfikatu serwera:

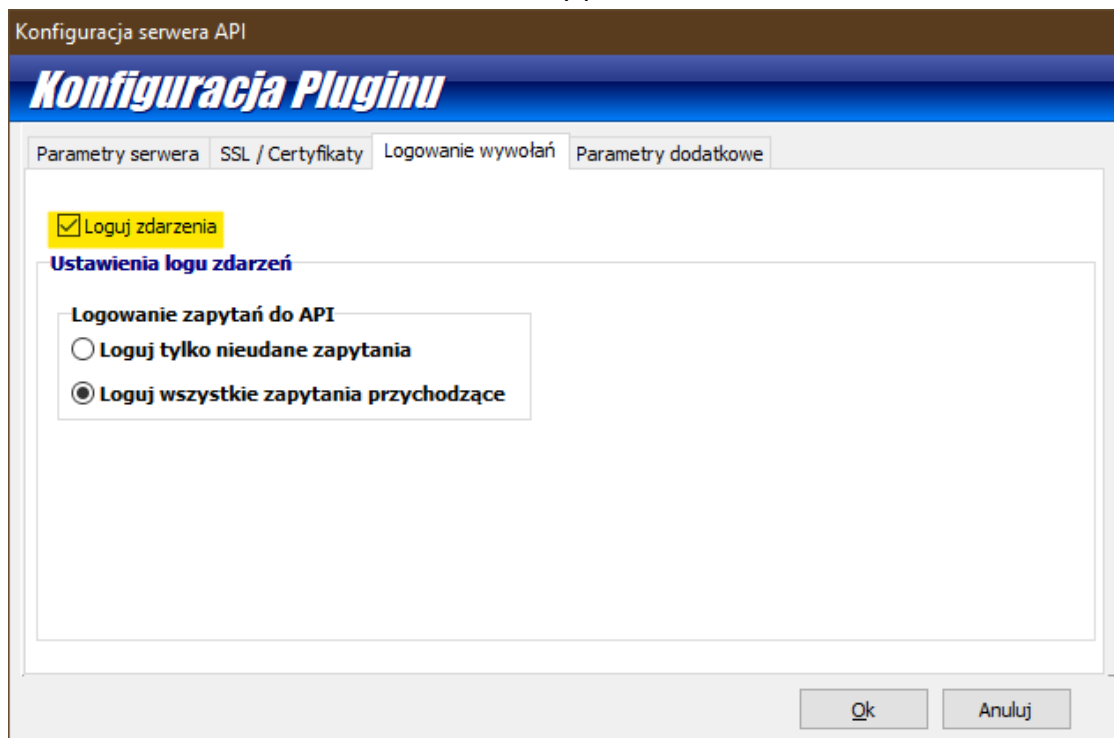
Plik root CA:

Hasło klucza serwera:

Plik klucza powinien być w formacie .pem, a plik certyfikatu w formacie .key. Plik root CA nie jest wymagany.

Ok Anuluj

- *Używaj połączenia SSL (Use SSL connection)* – check this option and select a certificate to use a secure connection to the API.
 - *Plik klucza serwera* – a server key in the PEM file format;
 - *Plik certyfikatu serwera* – a server certificate in the KEY file format;
 - *Plik root CA* – a root CA file (optional);
 - *Hasło klucza serwera* – a server key password;



Konfiguracja serwera API

Konfiguracja Pluginu

Parametry serwera | SSL / Certyfikaty | **Logowanie wywołań** | Parametry dodatkowe

Loguj zdarzenia

Ustawienia logu zdarzeń

Logowanie zapytań do API

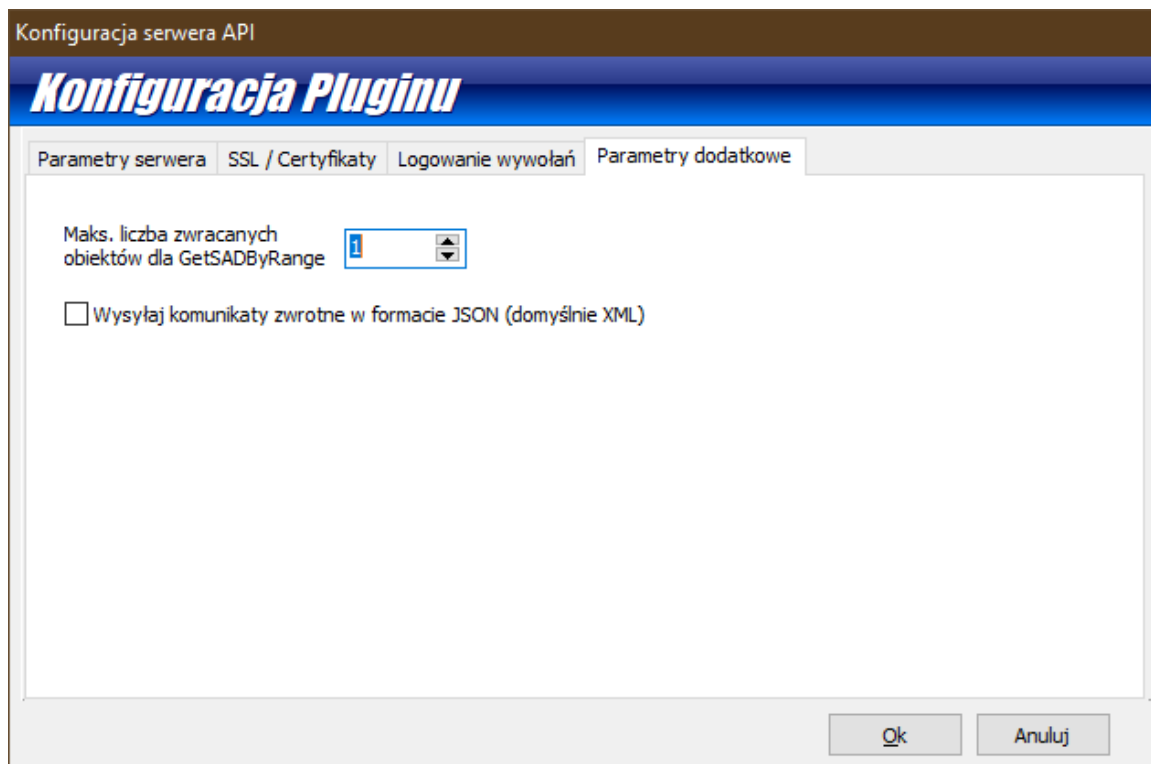
Loguj tylko nieudane zapytania

Loguj wszystkie zapytania przychodzące

Ok Anuluj

Events logging options:

- *Loguj zdarzenia (Log events)* – this option enables logging different events by the server. The options allow for logging only unsuccessful requests (*Loguj tylko nieudane zapytania*) or logging all the incoming requests (*Loguj wszystkie zapytania przychodzące*).



Additional options:

- *Maks. liczba zwracanych obiektów dla GetSADByRange (Max. number of objects returned by GetSADByRange)* – this option is used to restrict the number of objects returned by the GetSADByRange method (see the 5.1.4 section for more information).
- *Wysyłaj komunikaty zwrotne w formacie JSON (domyślnie XML) (Respond to the incoming requests in JSON format (default XML))* – this option allows the server to send responses in JSON format.

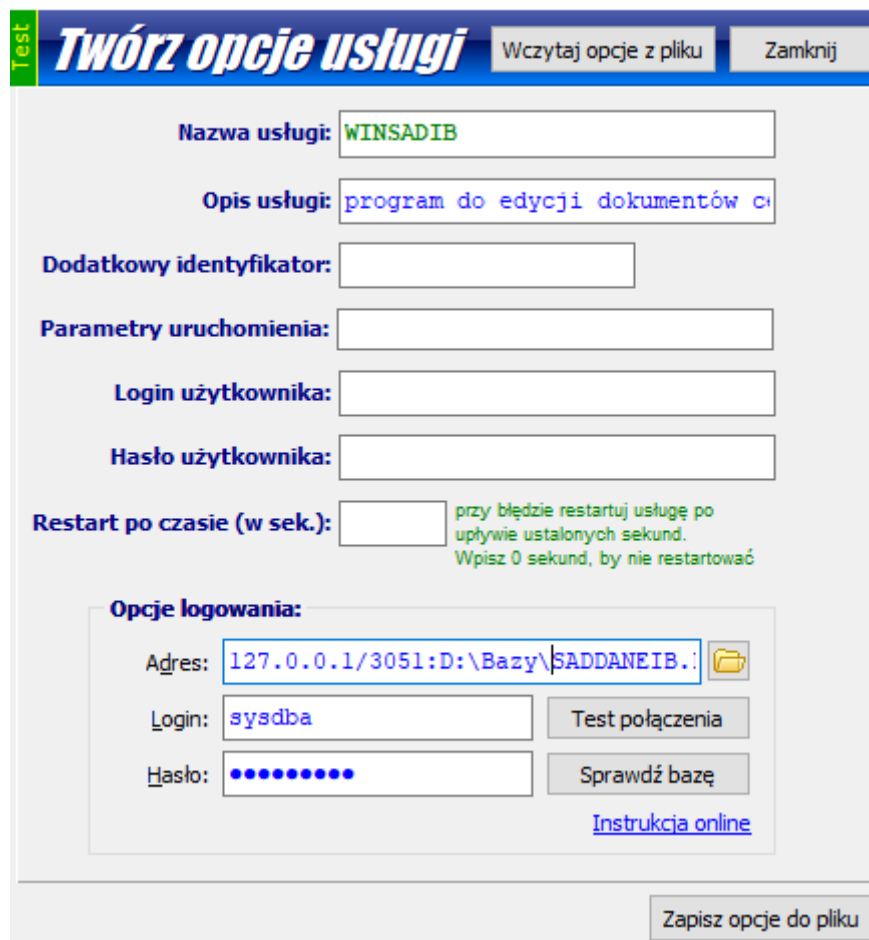
3.2. WINSAD SERVICE CONFIGURATION

Start WinSAD to configure it to work in service mode.

ATTENTION! The application working in service mode uses one license of the WinSAD application.

Select *Twórz opcje dla trybu usług WinSAD (Create configuration for WinSAD service)* from the *Narzędzia\Narzędzia administratora* menu.

Opcje usług



Twórz opcje usługi Wczytaj opcje z pliku Zamknij

Nazwa usługi:

Opis usługi:

Dodatkowy identyfikator:


Parametry uruchomienia:

Login użytkownika:

Hasło użytkownika:

Restart po czasie (w sek.): przy błędzie restartuj usługę po upływie ustalonych sekund.
Wpisz 0 sekund, by nie restartować

Opcje logowania:

Adres: 

Login: Test połączenia

Hasło: Sprawdz bazę

[Instrukcja online](#)

Zapisz opcje do pliku

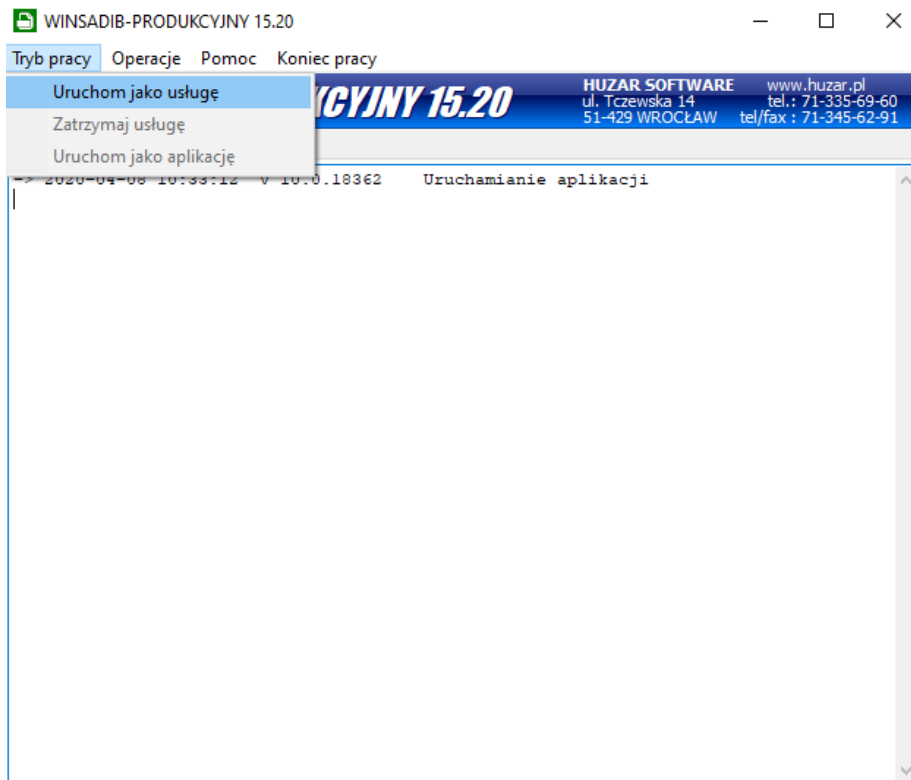
Parameters available in the configuration dialog:

- *Opis usługi (Service description)* – a description of the service to be displayed on the Windows services list.
- *Dodatkowy identyfikator (Additional identifier)* – fill this box if you intend to run more than one WinSAD service in a single Windows system environment; allows to distinguish the services.
- *Parametry uruchomienia (start-up parameters)* – a list of parameters used to start the service, same as the parameters used when running WinSAD in application mode.
- *Login użytkownika i Hasło użytkownika (User login and User password)* – credentials of the WinSAD user used to run WinSAD in service mode. It is recommended to create a dedicated WinSAD user account, which will act as a service account. When the service account is created, start WinSAD in application mode and log in as the service user to complete the configuration of the service user profile in WinSAD. **ATTENTION!** This is a user of Huzar Software application, not the Windows account used to run the service.

After saving the service configuration to a file, WinSAD will suggest creating a desktop shortcut to the WinSAD service management console.

3.3. *RUNNING WINSAD AS A SERVICE*

Using the shortcut created in 3.2, start the WinSAD service management console as system administrator and select *Uruchom jako usługę* from the *Tryb pracy* menu.



In the *Dane usługi NT* dialog, which will be displayed, select the user account to be used to start the WinSAD service. If the privileges of the Windows technical accounts are not sufficient, create a new user account in the system, select the *Inne konto* option and complete the *Nazwa konta (Account name)* and *Hasło dostępu (Password)* fields.

3.4. *HUZARAPI INSTALLATION TEST*

To check if the service is working properly, start any web browser and enter a sample query:

<http://127.0.0.1:3012/~HUZARAPI/GETSAD?IDSAD=1>

The above request works for a sample configuration specified in section 3.1. You should adjust it to your own configuration by entering the correct protocol (http or https) and port number. The elements of the address to be customized are underlined.

A properly configured and running API should result in getting the above request:

Returns a message if the SAD of ID=1 does not exist in the WinSAD application database

Brak SAD-u o ID: 1

or the SAD of ID=1 in WinSAD application database (if it exists) in DeclarationList format.

5. HUZARAPI WITH HTTP PROTOCOL

The API allows you to call a specific set of functions of WinSAD application. To obtain data from WinSAD database, http requests use GET (reading data from the database) and POST (writing data to the database) methods.

If # or @ special characters are present in the request parameters, they must be replaced by the corresponding codes:

- instead of # the code **%23** should be used
- instead of @ the code **%40** should be used

The # and @ characters may occur in document GUIDs requested from the database.

4.1. *METHODS USING HTTP GET*

Requests using GET method are performed by calling specific functions with parameters in HuzarAPI. Below you will find the list of available functions.

5.1.1. *GETSAD*

Function description:

Reads a SAD document from the database based on the given criteria (parameters).

Input parameters:

- IDSAD type: integer
- DOCREFNO type: text
- GUID type: text

Output:

XML / JSON object of DeclarationList type or an error message.

5.1.2. *GETSADSTATUS*

Function description:

Reads the status of a SAD document based on the given criteria.

Input parameters:

- IDSAD type: integer
- DOCREFNO type: text
- GUID type: text

Output:

The status of the SAD as an integer value.

Possible SAD statuses:

-1 – Unknown – returned in case of error or unsupported status;

10 – Preparing – a document in preparation – it has been created and may be modified by the user;

20 – Sent – the document has been sent to the customs system and is awaiting approval;

30 – Confirmed – the document has been approved by the customs and is awaiting clearance;

40 – Control – the document has been detained for customs control by the customs office;

50 – Rejected – the document has been rejected by the customs office;

60 – Annulled – the document has been annulled on the request of the sender;

70 – Cleared – the document has been cleared to the requested customs procedure;

80 – Closed – the document has been closed (for export IE599 and for transit IE029 has been received).

5.1.3. COPYSAD

Function description:

Creates a copy of the document indicated as a source in the input parameter. The copy is created according to the options of the WinSAD user whose account is used for the HuzarAPI service.

Input parameters:

- IDSAD type: integer
- DOCREFNO type: text
- GUID type: text

Output:

An XML / JSON object of DeclarationList type containing data of the newly created document.

5.1.4. GETSADBYRANGE

Function description:

Reads the list of SADs using the given criteria.

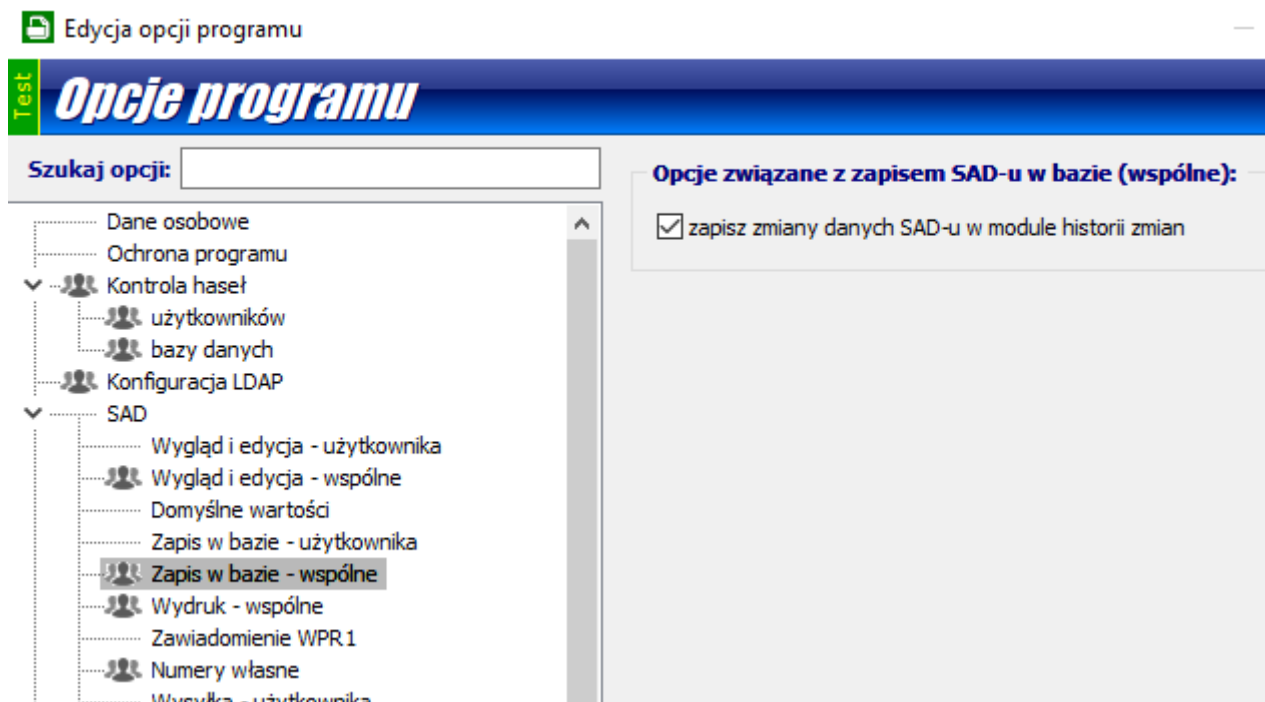
Input parameters:

- FROM type: date – date of the SAD document (start date)
- TO type: date – date of the SAD document (end date)
- BRANCHID type: text [3] (3 characters) – the ID of the branch to which the SAD is assigned
- LASTCHANGEID type: integer – identifier of the last change in the SAD document, all documents for which the change id is higher than the given parameter will be returned.
- DECLTYPE type: text [1] (declaration type – “I” import/ “E” export/” T” transit
- HEADERONLY type: boolean – possible values “true” or “false” – if the parameter is not given the value is defaulted as “false”. The parameter indicates whether the header data only or the complete declaration is to be returned (without or with declaration items and invoice items).
- SADSTATUS type: integer – code based on SAD document’s status; possible codes are described in section 4.1.2.

Output:

An XML / JSON object of DeclarationList type containing all the declarations matching the criteria given in input. In the Declaration.HSChangeID attribute returned in DeclarationList there is a new value of the last change id returned. HSChangeID is not filled, if it has not been given in input parameters.

To use the functionality of last change id, you need to enable logging of changes in WinSAD. To do that, from WinSAD main menu, select *Opcje\Opcje programu*, go to the *Zapis w bazie – wspólne* section and check the *Zapisz zmiany danych SAD-u w module historii zmian* option.



From that moment on, every change in the SAD documents will be logged in the database and you will be able to use the last change id as the function parameter.

Due to the possibility of including multiple SAD documents in the output of the function, the number of returned documents may be limited in the module options to avoid large amount of data contained in a single request. To do that, in the configuration of the HuzarAPI module, you should specify the value of the *Maks. liczba zwracanych obiektów dla GetSADByRange* parameter.

5.1.5. GETATTACHEDDOCLIST

ATTENTION! The GETATTACHEDDOCLIST function can only be used if the following conditions are met:

- An additional module called **Składnica Dokumentów** (Document Storage) is enabled and configured in WinSAD,
- Saving of customs documents to Składnica Dokumentów and attaching them to SADs is enabled in WinSAD configuration.

To obtain access to module Składnica Dokumentów please contact Huzar Software.

Function description:

Gets a list of documents attached to the SAD document. Documents are stored on 2 levels in WinSAD: documents and attachments. The list in the output contains a flat list of all attachments for a given SAD.

In case of multi-set SAD documents, each set will contain the same list of documents (there is no information to which set a document is attached to).

Input parameters:

- IDSAD type: integer
- DOCREFNO type: text
- GUID type: text

Output:

An XML / JSON object of AttachedDocumentList type containing a list of all attachment headers for a given SAD document.

5.1.6. GETATTACHEDDOCDET

Function description:

Gets a specific document attached to the SAD. The return value of the function contains a base64 encoded string in AttachedDocument type.

Input parameters:

- GUID type: text (GUID of the document)

Output:

An XML / JSON object of AttachedDocumentList type containing a single document encoded in base64.

5.1.7. GETCUSTOMSMSGLIST

Function description:

Gets customs messages attached to the SAD. The return value of the function contains a base64 encoded string in ListCustomsMSG type.

Input parameters:

- IDSAD type: integer
- DOCREFNO type: text
- GUID type: text (GUID of the document)
- MRN type: text
- MSGTYPE type: integer (list of values attached separately)

Output:

An XML / JSON object of ListCustomsMSG type containing a list of CustomsMSG, which holds a document encoded in base64. Additionally, the type holds basic information about the

document, namely: CustomsRefNo (customs reference number), IDSAD (SAD ID), MSGType (document type).

5.1.8. *GETGENERATEDDOCLIST*

Function description:

Gets a list of documents generated from the SAD. The return value of the function contains document ID, type, and number in ListGeneratedDoc type.

Input parameters:

- IDSAD type: integer
- DOCREFNO type: text
- GUID type: text (GUID of the document)
- MRN type: text
- MSGTYPE type: integer
 0. CMR
 1. EUR.1
 2. Cert. of origin
 3. A.TR

Output:

An XML / JSON object of ListGeneratedDoc type containing a list of GeneratedDoc, which holds values DocID (document ID), DocType (document type), and DocNo (document number).

5.1.9. *GETREPRESENTATION*

Function description:

Gets representations specified by provided parameters. The return value of the function contains all found representations as a list of RepresentationList type.

Input parameters:

- NIP type: string
- LASTCHANGEID type: integer
- TYPE type: integer
- TYPEOFVALIDITY type: integer
- ONLYVALID type: boolean

At least one of the parameters [NIP, LASTCHANGEID] is required.

Values for fields TYPE and TYPEOFVALIDITY are listed in attached excel file "HuzarAPI_Formats_v_X_XX.xlsx".

Output:

An XML / JSON object of RepresentationList type containing a list of Representation type, which holds essential information such as: number, type, type of validity, validity timeframe, revoked/deleted flags, last change date and ID, company.

5.1.10. GETDELIVERIES

Function description:

Gets MagCS deliveries specified by provided parameters. The return value of the function contains all found deliveries as a list of DeliveryList type.

Input parameters:

- BRANCHID type: string
- DELIVERYID type: integer
- LASTCHANGEID type: integer
- FROM type: date
- TO type: date
- HEADERONLY type: boolean

At least one of the parameters [BRANCHID, DELIVERYID, LASTCHANGEID] is required.

If HEADERONLY is set to true, then the function will omit listing shipments.

Output:

An XML / JSON object of DeliveryList type containing a list of Delivery type, which holds essential information such as: UID, delivery number, date of arrival, LastChangeID, deletion status, transportation info, and conditionally deliveries' shipments.

5.1.11. SENDSAD

Function description:

Submits a task of sending a declaration to the polish customs system. The return value of the function informs whether the task was scheduled or not.

Input parameters:

- IDSAD type: integer
- DOCREFNO type: text
- GUID type: text (GUID of the document)
- MRN type: text
- USER type: string
- EXPORTTYPE type: integer

0. Webservice channel
 1. E-mail channel
 - GATEWAY type: integer
 1. PUESC v1
 2. PUESC v2
 3. BCP
 4. BCP e-Commerce
 - THUMB type : string

Required parameters: one of [IDSAD, DOCREFNO, GUID, MRN]; USER; EXPORTTYPE; GATEWAY.

Output:

Declaration ID if the task was scheduled successfully, an error message otherwise.

4.2. *METHODS USING HTTP POST*

For functions using the http POST method, the parameter(s) are enclosed in the BODY element of the request.

5.2.1. ADDSAD

Function description:

Adds new documents to WinSAD database. Some data will be deleted before saving the document to meet the technical requirements of a new document in WinSAD.

Input parameters:

- BODY type: DeclarationList

Output:

The request returns a DeclarationList object completed with technical data (Huzar dbID, GUID of the document, etc.)

Remarks:

- The SAD document can be created in two different modes – based on given *DeclItems* (items of the SAD declaration) or *Invoices* (invoices and invoice items). In one function call it is not possible to give both types of lists – if they are given simultaneously an error will be returned. The SAD created based on invoices will be created in the “based on invoices” mode and items of the customs declaration will be created based on parameters given in *GroupingOptions* node. If *DeclItems* are enclosed in the message, the document will be created in “manual” mode and the declaration items will exactly reflect the data sent in the input parameter to the function.

- In the “based on invoices” mode, the *ItemSyncOptions* node can be added which will cause synchronization of invoice items with items master data in WinSAD.
- You can attach document scans (like invoices, specifications, proofs) to the SAD document – in the , Attachments
 - DocID
 - DocMergelD
 - Name
 - Extension
 - DocType
 - DocContent

M an..35 Unique identifier of the attachment

O an Used for merging various documents under the same identifier into one

M an File name of the attachment

M an Attachment's extension - necessary for further preview in WinSAD

O an

M an Base64 encoded document

6. DATA FORMATS

In order to communicate with external systems, appropriate data formats have been prepared. These formats are mapped to objects used internally in Huzar Software applications. The structure and description of the possible messages can be found in the attached HuzarAPI_Formats_v_X_XX.xlsx file.

5.1. *CUSTOMS DECLARATION LIST*

This format is used to transfer customs declaration data. By default, it is always sent as a list of documents. If a Huzar system document is a multiset document (i.e. the declaration created on the basis of an invoice contains more declaration items than the maximum number allowed for the customs system), the return value for the function will contain multiple declarations with a single Huzar Software identifier.

5.2. *DOCUMENTS ATTACHED TO CUSTOMS DECLARATION*

Format containing details of documents attached to the customs document. These are usually documents read from the Składnica Dokumentów module. In case of proper configuration of WinSAD and Składnica Dokumentów (described above) it is possible to obtain the customs messages that are related to a SAD document.

7. EXAMPLES

Below you can find examples of requests and their descriptions:

6.1. *GETSAD*

7.1.1. *DECLARATION WITH A CERTAIN ID*

Request returning the customs declaration with WinSAD database ID 55753

<http://127.0.0.1:3012/~HUZARAPI/GetSAD?IDSAD=55753>

If the declaration exists, the return value will be enclosed in the BODY element of the message as an XML / JSON e.g.:



GETSAD_Response.
xml

7.1.2. *ERROR MESSAGE*

If the SAD document does not exist, the BODY element contains the message:

Brak SAD-u o ID: 55753

6.2. *GETSADBYRANGE*

6.2.1 *DATE RANGE*

The below request will return a list of customs documents from the given date range – in this example declarations from 2020.04.01 to 2020.04.20

<http://127.0.0.1:3012/~HUZARAPI/GetSADByRange?FROM=20200401&TO=20200420>

In the BODY element of the message you will get XML data e.g.:



GETSAD_ByRange_1
.xml

6.2.2 *HEADERS*

To obtain document headers only, without detailed information of declaration items, the request for the same date range as above is as follows:

<http://127.0.0.1:3012/~HUZARAPI/GetSADByRange?FROM=20200401&TO=20200420&HEAD-ERONLY=True>

In the BODY element of the message the response contains XML / JSON data e.g.:



GETSADByRange_2.
xml

6.2.3 LAST CHANGES

To get a list of last changes for customs declarations, you have to enable the history of changes in WinSAD (as described above). In the input parameter, enter the identifier of the last change that was registered in your system (all changes with the higher change ID will be returned)

<http://127.0.0.1:3012/~HUZARAPI/GetSADByRange?LASTCHANGEID=19587>

In the BODY element of the message the response contains XML / JSON data e.g.:



GETSADByRange_3.
xml

Please note that each document on the list will have the same HSDBChangeID, which refers to the last change returned. In case of subsequent requests, this identifier must be entered in the next request to obtain the next set of database changes.

6.2.4 ERROR MESSAGE

If the list is empty, the BODY part will contain the text:

Nie udało się wczytać obiektów SAD

6.3. GETSADSTATUS

6.3.1. REQUEST ON THE SAD STATUS

Request on the SAD status with customs reference number 19S116010U:

<http://127.0.0.1:3012/~HUZARAPI/GetSADSTATUS?DOCREFNO=19S116010U>

In the BODY part of the message you will get a return value as below:



GETSADStatus.xml

6.4. *GETCUSTOMSMSGLIST*

6.4.1 *REQUEST ON THE SAD CUSTOMS MSG LIST*

Request on the SAD customs MSG list with customs reference number 19S116O10U:

<http://127.0.0.1:3012/~HUZARAPI/GetCUSTOMSMSGLIST?DOCREFNO=19S116O10U>

In the BODY part of the message you will get a return value as below:



GetCustomsMSGList.
json

6.5. *GETGENERATEDDOCLIST*

6.5.1 *REQUEST ON THE SAD GENERATED DOCS*

Request on the SAD generated documents list with customs reference number 19S116O10U:

<http://127.0.0.1:3012/~HUZARAPI/GetGENERATEDDOCLIST?DOCREFNO=19S116O10U>

In the BODY part of the message you will get a return value as below:



GetGeneratedDocList.
json

6.6. *GETREPRESENTATION*

6.6.1 *REQUEST ON THE COMPANY'S REPRESENTATIONS*

Request on the company's representation list with NIP number 1234567890:

<http://127.0.0.1:6000/~HUZARAPI/GETREPRESENTATION?NIP=1234567890>

In the BODY part of the message you will get a return value as below:



GetRepresentation.
xml

6.7. *GETDELIVERIES*

6.7.1 *REQUEST ON MAGCS DELIVERIES*

Request on the delivery list with LASTCHANGEID equal to 0:

<http://127.0.0.1:6000/~HUZARAPI/GETDELIVERIES?LastChangeID=0>

In the BODY part of the message you will get a return value as below:



GetDeliveries.xml

6.8 SENDSAD

6.8.1 REQUEST ON SENDING A DECLARATION

Request on sending a declaration of ID 3469 through the e-mail channel and BCP gate via user Test1:

<http://127.0.0.1:6000/~HUZARAPI/SENDSAD?IDSAD=3469&USER=Test1&EXPORTTYPE=1&GATEWAY=3>

In the BODY part of the message you will get a return value as below:



SendSAD.xml







6.9 ADDSAD

6.9.1 REQUEST ON ADDING A DECLARATION

Request on adding a declaration:

<http://127.0.0.1:6000/~HUZARAPI/ADDSAD>

In the request BODY a document in the *Declaration* format has to be sent (examples below):

SAD [manual]	 Declaration.xml	SAD AIS H7 [manual]	 Declaration H7.xml
SAD with company DB [manual]	 Declaration with company DB.xml	SAD AIS H7 OISS [manual]	 Declaration H7 IOSS.xml
SAD [invoice based]	 Declaration invoices.xml	SAD AIS H7 OISS [invoice based]	 Declaration H7 IOSS invoices.xml

In the BODY part of the message you will get a return value as below:



AddSAD.xml