TSS Interface description

Version 1.04

NTDSC TSS data provision interface description

03/10/2025

Hungarian Tourism Agency

Document control

Version	Phase	Published	Responsible person	Reason/Changes
v1.00	Accepted version	04/04/2022	HTA	First published version
v1.01	Updated version	20/08/2022	HTA	Second published version
v1.02	Updated version	20/09/2022	HTA	Expand interface based on feedback, transfer field-level specifications from Excel attachment
V1.03	Updated version	27/02/2025	HTA	Specification of required data transmission frequency, field comment clarifications.
v1.04	Updated version	03/10/2025	HTA	Additional field validation rules, developer comments and suggestions added, value set of fields *Reason for combined data provision* (osszesitettIndoklasa) and *Ticket validity type* (jegyErvenyessegTipusa) expanded.

CHANGE LOG **5**

GLOSSARY **7**

1. INTRODUCTION **8**

1.1. THE PURPOSE OF THE DOCUMENT 8
1.2. TECHNICAL DESCRIPTION OF THE TSS DATA PROVISION INTERFACE 9

2. THE TSS SOFTWARE CONNECTION PROCESS **10**

2.1. SUBMITTING A TSS SOFTWARE CONNECTION REQUEST 11
2.2. PREPARING THE TEST ENVIRONMENT FOR TSS SYSTEM CONNECTION 12
2.3. GENERATE AND SEND DIGITAL CERTIFICATE SIGNING REQUESTS 12
2.4. PROVISION OF INFORMATION NEEDED TO CONNECT TO THE TSS SYSTEM 12
2.5. TSS SYSTEM DEVELOPMENT 15
2.6. CONDUCTING TESTING IN THE TEST ENVIRONMENT OF THE NTDSC ATTRAKCIÓ SYSTEM 15
2.7. ASSESSMENT OF VALIDATION TESTS 15
2.8. ISSUING A CERTIFICATE 15

3. CONFIGURATION FOR LIVE DATA TRANSMISSION **16**

3.1. THE STORAGE OF THE TSS SOFTWARE IDENTIFIER IN THE NTDSC ATTRAKCIÓ SYSTEM 16
3.2. REGISTRATION OF SERVICE PROVIDER 16
3.3. SERVICE LOCATION REGISTRATION 17
3.4. ISSUE CERTIFICATE AND ASSIGN SERVICE LOCATION TO THE TSS SYSTEM 17
3.5. TSS SYSTEM CONFIGURATION IN A LIVE ENVIRONMENT 18
3.6. COMMENCEMENT OF LIVE DATA TRANSMISSION 18

4. TSS MESSAGE TYPES, MESSAGING PROCESS **19**

4.1. SUMMARY 19
4.2. ORDER OF DATA SUBMISSIONS 20
4.3. SUBMIT TICKET SALES DATA 20
4.4. SUBMIT TICKET VALIDATION DATA 23
4.5. VERIFICATION OF SUBMITTED DATA 23
4.6. SPECIFICATION DESCRIPTION OF MESSAGES 24
4.6.1. Structure of message header 24
4.6.2. Ticket sales message structure 27
4.6.3. Ticket validation message structure 53
4.6.4. Verification request message structure 59
4.6.5. Response messages 60

5. FORMAL REQUIREMENTS **62**

5.1. GENERAL DATA SUBMISSION REQUIREMENTS 62
5.2. ERROR KEYS ON THE TSS INTERFACE 65

6. APPENDICES **67**

6.1. REQUESTING A CERTIFICATE 67
6.1.1. Standards and Recommendations 67
6.1.2. Creating a certificate signing request in a Windows environment 67
6.1.3. Creating a certificate signing request in a Linux environment 68

6.2. INTERFACE MESSAGE EXAMPLES 70
6.2.1. Sales for an independent programme 70
6.2.2. Sales for a combined programme 72
6.2.3. Ticket validation 75
6.2.4. Verification 76
6.3. COUNTRY CODES 78
6.4. PROGRAMME CATEGORIES 84

Change log

Changes introduced in version 1.04

- The earliest year that can be entered in the date fields has been set to 1800.
- Negative values have been excluded from all date fields.
- A továbbiakban önálló programtípussal ellátott programok esetén nem küldhető érték a kapcsolódó programok mezőben. (4.6.2.1.2.)
- In the future, for programs with a separate program type, no value can be entered in the related programs field. (4.6.2.1.2.)
- A detailed description of the error messages to be returned in the absence of mandatory data blocks has been added.
- Verification of submitted data – processing results available at the verification endpoint (4.5)The accepted value sets for the following two fields have been extended:
    - *Ticket sales message / Transaction data / Reason for combined data provision* (osszesitettIndoklasa*)*
    - *Ticket sales message / Transaction data / Tickets sold / Ticket validity type (* jegyErvenyessegTipusa)

Changes introduced in version 1.03

- The data transmission frequency has been specified, with an initial value of 15 minutes.
- It has also been documented that it is mandatory to send transactions in batches in order to reduce message traffic.
- Clarification of the comment in the field *Program identifier at the service location* (szolgHelyProgramAzonosito)

Changes introduced in version 1.02

- A recommendation for age groups has been added in the comment field of the *Age Group* field.
- TSS system developers should prepare their application(s) for possible future changes in the set of values of the fields containing ENUMs defined in the interface
- *Ticket sales message / Programme data / Online programme* field was added.
- The field *Ticket sales message / Transaction data / Transaction total* has been added
- *Ticket sales message / Transaction data / Final discount* has been deleted
- The value set of the *Ticket sales message / Transaction data / Other items / Category* field has been updated:
    - New item
        - Discount
- The *Ticket sales message / Transaction details / Payment method / Amount paid in HUF* field type has been changed to integer (int)
- Tthe JSON name of the *Combined data provision* field has been changed. New name: *osszesitett.*

- The JSON name of the field *Other items* has been changed to *egyebTetelek*.
- The validation of the field *Transaction data / Other items / Quantity* has been updated.
- The type of the field *Ticket sales message / Tickets sold / Gross price* has been changed to integer (int).
- The JSON name of the field *Transaction Data / Other Items / Gross unit price* has been changed. The new name is*: bruttoEgysegAr*
- The field *Ticket validation message / Validation identifier* has been added.
- Updated the value set of the *Transaction data / Means of payment / Payment Method* field:
    - New item
        - Rounding (KEREKITES)
- The rules for calculating the rounding payment method were added to the document.

Glossary

This chapter describes the terms and definitions that are necessary to understand and implement the data reception interface.

Term, abbreviation	Definition
CSR - Certificate Signing Request	Certificate Signing Request. To provide data in the NTDSC system, the use of a unique electronic certificate is required to identify the tourist attraction and the software instance used there. The user representing the service provider, with the support of the TSS system vendor, creates Certificate Signing Requests (CSRs) which are uploaded to the NTDSC.
Tickets sold	Tickets sold for a specific occasion of a specific programme.
Ticket type	Generic ticket categories based on different ticket characteristics (related age group, discounts, default validity, etc.), which are used as templates when pricing programmes.
JKÜ	Authorization Manager Agent (JKÜ), a central service of NISZ Zrt., which allows to check the permissions of the logged-in person to manage transactions for a given tax number (for a registered company or sole trader).
HTA	Hungarian Tourism Agency
NTDSC ATTRAKCIÓ portal	The NTDSC ATTRAKCIÓ system is the portal interface of the system where service providers of tourist attractions can register.
NTDSC ATTRAKCIÓ system	A system for collecting statistical information on ticket sales and usage sent by service providers.
Programme	A permanent, periodic or fixed-date event or essential service for which tickets are sold.
ROA	Registry of authorizations (https://rendelkezes.gov.hu/). A person authorized for representing a given service provider may use this to authorize another person to act in cases related to the service provider in the NTDSC system.
Service location	An independent programme management and ticket sales unit of a service provider. Not the same as the venue. E.g.: Hungarian National Museum Esztergom Castle Museum
Service provider	Organiser of a tourism-related programme (see programme definition). In the NTDSC ATTRAKCIÓ system, the representative or authorised representative of the service provider is entitled to register the service provider.
Certificate	One form of electronic signature (https://en.wikipedia.org/wiki/Electronic_signature), used by the system to authenticate messages exchanged between TSSs and the NTDSC ATTRAKCIÓ system, or requested by the sending endpoints.
TSS	Ticket Sales Systems*(Ticket Sales System*)
TSS interface (TSS IF)	The interface for data transmission provided by the NTDSC ATTRAKCIÓ system.
Purchase transaction	A particular sales transaction that may involve multiple tickets.

1. Introduction

The National Tourism Data Supply Centre (NTDSC) supports the digitalization, development and whitening of the tourism industry and is therefore indispensable for improving the competitiveness of the industry. Currently, not every enterprise directly associated with tourism is included in the system, therefore, widening the scope is in the interest of the national economy. Potential sources of some of the missing data could be the Ticket Sales Systems used by individual tourist attractions - be it visitor numbers, demographic characteristics, spending, seasonality, or back-testing the success of promotions.

In the framework of the further development of the NTDSC system, the unified registration, data provision and statistical data management of tourist attractions in Hungary will be implemented in addition to the analysis of the turnover data of accommodation establishments. The NTDSC ATTRAKCIÓ system is part of the NTDSC IT system, which receives data from tourist attractions, stores and processes them, and produces structured reports, risk analyses and provides them to the sector's data providers and the statutory authorities.

In order to meet the defined objectives, a TSS (Ticket Sales System) data provision interface will be developed within the NTDSC ATTRAKCIÓ system, through which tourist attractions will be able to submit their statistical data to the NTDSC system.

1.1. The purpose of the document

This document details the prerequisites for the integration of the NTDSC ATTRAKCIÓ system with the Ticket Sales Software System (hereinafter referred to as TSS), the configuration of the integration on the TSS side, its communication channels and the rules for data transmission. This document is not intended to provide guidance on the internal data structure, architecture and modules of TSS software, but only the information necessary for interfacing with the NTDSC ATTRAKCIÓ system is provided.

This document contains the following chapters:

2. TSS software integration process,
    a. this chapter describes the connection process step by step and what each player has to do at each step.
3. Configuration for live data transmission
    a. This chapter describes how, after a successful connection, the service provider and the TSS vendor can prepare for live data transmission.
4. Types of TSS messages
    a. Data on 2 data sets are required as part of the data provision:
        i. ticket sales
        ii. ticket validation

        In this chapter you will find a detailed description of these message types and access the message specification description.
5. Formal requirements
    a. This chapter contains the formal requirements for messages

1.2. Technical description of the TSS data provision interface

The TSS interface is a REST-based, data receiving service that processes the submitted data asynchronously to ensure high load capacity.

Submitted messages must be signed by the sending TSS system with the NTDSC TSS certificate requested and downloaded from the NTDSC interface, thus ensuring the authenticity and integrity of the data. The same certificate is used to authenticate and identify the TSS instance during the TLS/SSL connection establishment.

The submitted data will be received by a front-end application that will check the digital signature of the message. Afterwards, it will be checked whether the data submitted contain the fields corresponding to the interface description, and whether every mandatory field has been filled out. If a message is acceptable, the front-end application will generate a **unique processing identifier** which the calling system will receive immediately in the response. The front-end application will forward the message received to NTDSC core systems for processing. This identifier can be used subsequently to check status of the processing of the message submitted.

The messages will then be forwarded to the NTDSC ATTRAKCIÓ system for processing.

The benefits of this solution:

- due to asynchronous processing, the TSS interface can serve a larger workload than services implementing synchronous processing;
- with the help of the verification endpoint, the call systems may query the final processing result of the data submitted. This ensures that the data have been properly received and processed;
- asynchronous processing allows for more complicated, albeit more time-consuming, checks to be performed on the messages received.

2. The TSS software connection process

In this chapter, the steps to integrate the TSS software for data transmission with the NTDSC ATTRAKCIÓ system are described.

The flowchart below indicates the steps required for a successful connection. The steps in the process shown in the diagram are explained in detail below.

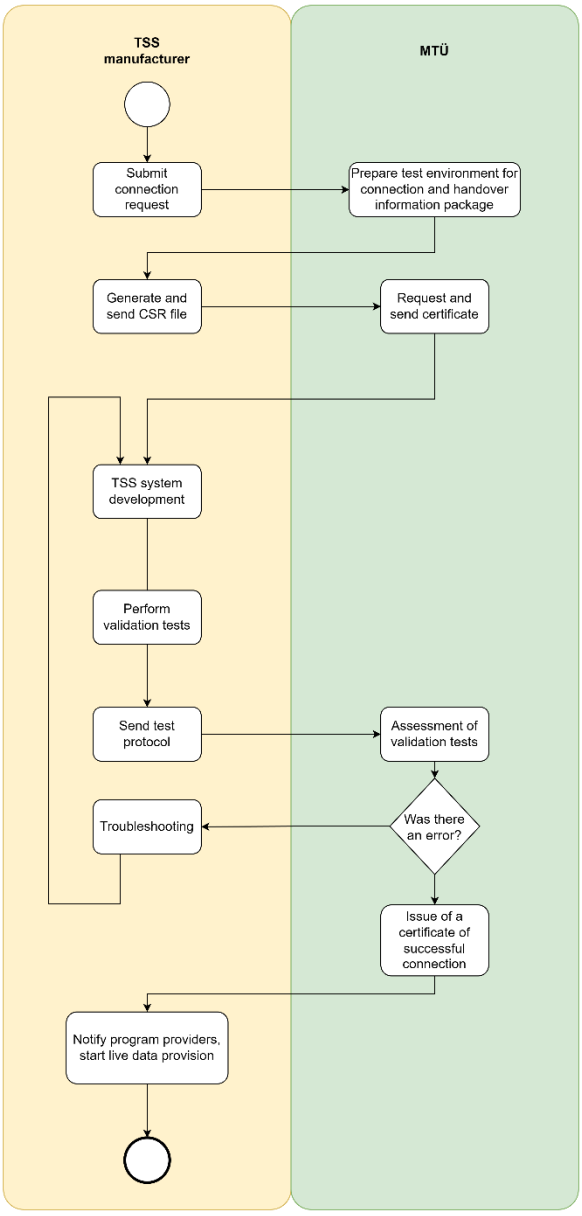


Figure 1 - TSS system connection flowchart

Steps in the TSS software connection process:

a. Submitting the connection request – Responsible Person: TSS manufacturer
b. Prepare test environment and send information package and reference client to TSS manufacturer to support integration - Responsible: HTA
c. Generating the CSR files necessary for requesting the certificates – Responsible Person: TSS manufacturer
d. Requesting certificates in the testing environment of the NTDSC ATTRAKCIÓ system - Responsible: HTA
e. TSS system development - Responsible: TSS manufacturer
f. Performing validation tests and sending test report – Responsible Person: TSS manufacturer
g. Assessment of validation tests – Responsible Person: HTA
h. Troubleshooting if necessary - Responsible: TSS manufacturer
i. If tests are successful, issuing a certificate to verify successful connection – Responsible Person: HTA
j. Notify program providers, start live data provision - Responsible: TSS manufacturer

2.1. Submitting a TSS software connection request

The service locations registering in the NTDSC ATTRAKCIÓ system can choose during the registration process which TSS system or systems they wish to use for their data provision. In order for a TSS to be included among the TSSs that can be selected during the NTDSC registration process, the owner/distributor of the TSS software must submit a pre-registration application with the basic TSS software data to the HTA.

Data to be provided in advance:

- TSS manufacturer data (company data, contact details, public data);
- IP address or domain of the TSS system developer and test environment;
- the scheduled start date of the validation test.

You can download the joining form from the link below. The completed form can be sent to dev.support@ntak.hu e-mail address.



Download connection form

2.2. Preparing the test environment for TSS system connection

After the acceptance of the connection form, the HTA will perform the following settings in the test environment of the NTDSC ATTRAKCIÓ system in order to allow the TSS system manufacturer to start the integration.

Includes the TSS system in the NTDSC ATTRAKCIÓ system as one of the Ticket Sales Systems that can be selected by the service providers. You also register a test provider and test service location for the connection process. These data, including the tax number of the service provider and the registration number of the service location, will be sent to the TSS system manufacturer. The service location registration number will be required to generate the digital *Certificate Signing Request* (CSR) for certificate enrollment.

2.3. Generate and send digital certificate signing requests

The next step in the process is to generate a digital certificate signing request for certificate enrollment, for help see chapter 6.1.

It is important to note that the CSR file must be generated at the place where the data transmission will be sent from.



The CSR file should be sent by e-mail to the dev.support@ntak.hu e-mail address. The subject must indicate the name of the TSS system to be connected and the test service location identification number.

2.4. Provision of information needed to connect to the TSS system

Upon receipt of the CSR file, the HTA staff will request a certificate for the TSS manufacturer via the NTDSC ATTRAKCIÓ system interface.

Once the certificate is completed, the HTA will send the TSS software owner/distributor the information necessary to access the test environment and configure the data transmission.

These information are the following:

- Access data for the test system of the NTDSC ATTRAKCIÓ system;
- The certificate of the service location to be used for testing;
- Identification data required to test the TSS software;
- Swagger description URL;
- Reference client;
- Description of validation tests and test protocol template.

Access data for the test system of the NTDSC ATTRAKCIÓ system

The test system of the NTDSC ATTRAKCIÓ system paths contain the REST endpoints that can be used to test the data transmission and the test system path address. The HTA can provide the TSS manufacturer with access to the test system in two ways: With a VPN profile or with IP filtering.

If a VPN profile is used, the VPN profile included in the reply message needs to be correctly configured to access the environment. Instead of using a VPN profile, the HTA reserves the right to whitelist the IP domain specified by the TSS manufacturer and thus provide access to the test environment. The HTA will notify the TSS manufacturer by e-mail of the means of access to the TSS interface (VPN or IP filtering).

Service location certificate for testing

A prerequisite for data transmissions to the NTDSC ATTRAKCIÓ system's test environment is the use of a unique NTDSC TSS (test) certificate per service location to identify the sending party when establishing the SSL connection, and the electronic signature of the message and the attachment of the certificate to the message sent. Subsequent chapters will provide more detail on the use of the certificate.

Identification data for testing the TSS software

As part of the data transmission to the NTDSC ATTRAKCIÓ system, the TSS software is required to identify the sending service location, its service provider and the TSS software itself in the message body of the message sent.

These data will be sent by the HTA to the owner / distributor of the TSS software in a response letter to the TSS software connection request during the testing phase of the software.

These identification data are the following:

- **Service location registration number**: The identifier of the example service location that can be used for testing, containing ten uppercase letters and numbers. This data is required to be sent to the TSS interface in the field "szolgHelyRegisztraciosSzam".
- **Service provider's tax number:** The tax number of the service provider is the tax number used to identify the organisation belonging to the example service location. This data must be sent to the TSS interface in the field "adoszam".
- **TSS system identifier in the NTDSC system**: For testing and data provision, the TSS software needs to identify itself. This data must be sent to the TSS interface in the field "tssRendszerNTAKAzonosito".

Reference client

The reference client is an environment developed in Postman that allows you to understand in detail the structure of the messages to be sent. It also includes ready-made example messages to test and try out the data provision.

Description of validation tests

After the development of the TSS system, the manufacturer shall demonstrate by a validation test that the Ticket Sales Systems it has developed is able to send ticket sales data to the NTDSC ATTRAKCIÓ system correctly, to retrieve the results of the processing and to resend the corrected data in case of errors.

During the validation test, each ticket sales software vendor must perform the test cases defined by the HTA. During the validation test, all manufacturers must complete the minimum test cases. In addition, test cases have been identified that cover more complex feature sets, allowing the Ticket Sales System to provide additional functionality to tourist attractions that use its software. The description of each set of functions and test cases will be provided by the HTA as part of the information package.

Test report

During validation testing, the ticket sales software vendor must record the following data in the test protocol:

- Name of TSS software;
- TSS software identifier in the Test NTDSC system;
- identification data of the test service location;
- testing window;
- marking of test cases undertaken;
- processing identifiers received for ticket sales (and ticket validation, if any) messages;
- screenshots of the Ticket Sales System, showing that ticket sales have taken place on it;
- other relevant information.

A template for the test report will be provided by the HTA to the TSS manufacturer as part of the information package.

2.5. TSS system development

The next step in the process is to develop the TSS system. After the above configuration settings have been made, the TSS software can send data to the NTDSC ATTRAKCIÓ system's test environment, using any of the described endpoints.

During the development, the reference client can provide support, and support is also available at dev.support@ntak.hu.

Please indicate the name of the connected TSS system and the registration number of the test service location in the subject line of the e-mail.

2.6. Conducting testing in the test environment of the NTDSC ATTRAKCIÓ system

Data submissions will be subjected by HTA to technological validation tests and content verification, too.

During the validation test, the TSS manufacturer must indicate which requirements it undertakes to meet. These requirements will be the subject of the test and will determine which groups of functions will be indicated in the certificate in the event of a successful connection.

The validation test will be carried out in the time window chosen by the TSS manufacturer, within which the requirements agreed by the ticket sales software manufacturer must be met. During the validation tests, the required information must be recorded in the test report, the test report and the information to be recorded therein are included in the information package received during the connection. The test report is to be sent to the following e-mail address: dev.support@ntak.hu . The subject of the e-mail should indicate the name of the connecting Ticket Sales System and the registration number of the test service location used to connect.

2.7. Assessment of validation tests

HTA staff will examine the messages received for form and content.

If they find any errors, they will report them in writing to the manufacturer of the ticket sales software. Once the errors have been corrected, the process will resume from chapter 2.5.

2.8. Issuing a certificate

In case a given version of the TSS software passes the validation test, the HTA will issue a certificate to the user, stating for which functional groups the ticket sales software is capable of data transmission. The ticket sales software is then certified ready for live data submission.

3. Configuration for live data transmission

In case the TSS passes the required validation test, the HTA will add the TSS software to the TSS register of the live NTDSC ATTRAKCIÓ system and will send the live environment identifier of the TSS software to the TSS software vendor/distributor.

The service locations registering in the NTDSC ATTRAKCIÓ system can then select the TSS software for their data transmission.

3.1. The storage of the TSS software identifier in the NTDSC ATTRAKCIÓ system

The TSS software needs to store the TSS ID sent after successful validation. This is a string, independent of the identification service locations, which identifies the TSS software itself. The same identifier will be sent for each service location, for each data submission. The certificate used for data provision also contains, among other things, the TSS identifier.

Here are the things service providers should do. These consist of the following steps:

- NTDSC registration of the service provider
- Service location NTDSC registration;
- Specify the Ticket Sales Systems used;
- Request a digital certificate signing request from the TSS manufacturer;
- Uploading a digital certificate signing request to the NTDSC ATTRAKCIÓ system;
- Download the completed certificate and send it to the TSS manufacturer.

These steps are described in detail in the NTDSC ATTRAKCIÓ system user manual.

3.2. Registration of service provider

The prerequisite for data transmission is that the service provider who operates the tourist attraction registers in the NTDSC ATTRAKCIÓ system. The registration can be carried out by the person authorised to represent the service provider (or by the person authorised by the service provider in the ROA). It consists of the following steps:

1) Access the NTDSC Registration module interface. Login requires Client Gate identification.
2) Registration of service provider
    a. Access the interface: Service provider / New service provider menu.
    b. During the registration process, you must select the type of business organisation of the service provider to be registered, which can be a company, a sole trader or other organisation.
        i. For companies and sole traders, the system checks whether the logged-in user is authorised to act on behalf of the organisation, based on the Authorization Manager Agent (JKÜ) service.
        ii. For other organisations, this check will be done after the registration is completed, based on the documents uploaded during the registration process.

c. After a successful eligibility check, you will be asked to enter the name of the service provider and contact details. After this, press the Save button to complete the registration. (In the case of other organisations, the check will be carried out afterwards, and the user who initiated the registration will be informed of the result by e-mail.)

3.3. Service location registration

After the registration of the service provider, the registration of the service location(s) is also required. The service location is the service provider's independent programming unit. A service provider can have more than one service location. You can register a service location from the list page under the Service locations menu in the NTDSC registration module by clicking on the Add new service location button. When registering a service location, in addition to entering the required data, the user has the option to select the sector(s) of the NTDSC to which the service location will provide data. In the present case, the NTDSC ATTRAKCIÓ system should be selected. Once a service location has been registered, the user is returned to the Service Locations list page, where the newly added service location is displayed. The user also has the possibility to navigate to the NTDSC ATTRAKCIÓ system's interface with the click of a button. There is only one technical prerequisite left for data transmission: obtaining the necessary certificate for data transmission from the NTDSC ATTRAKCIÓ system. The following chapter will describe the relevant steps.

3.4. Issue certificate and assign service location to the TSS system

The ticket sales software used by service locations will attach an electronic signature to the submitted data during data provision. To create a trusted electronic signature, you need a certificate, which is provided by a *Certificate Authority* (CA).

1) **The TSS software vendor/provider is responsible for supporting the tourist attractions in generating the key pair and the certificate signing request (CSR) for issuing the certificate.**
    a. For guidance on the preparation of certificate signing requests, ticket sales software vendors are referred to chapter 6.1.2 of this document.
2) **Submitting the certificate signing request – requesting a certificate**

Service providers can apply for a certificate via the NTDSC ATTRAKCIÓ system and download the completed certificate from the same system. They can do this on the page they can access under menu item Select software, after the required software has been added, by clicking the button Certificate management in the appropriate line of the list. Certificate signing requests can be uploaded at any time - this means that the registration of a tourist attraction can be completed even if the required certificate is not yet available. A service location may (at least in principle) use more than one Ticket Sales System. The steps to apply for a certificate are as follows:

a) The service provider's representative uploads the requests generated by the ticket sales software (CSR file in the appropriate fields) to the certificate signing request interface found as described above.
b) The certificate signing request is submitted by clicking on the Send certificate signing request button. This is how requests are forwarded to the certificate authority (CA).

c) The NTDSC ATTRAKCIÓ system sends the request to the CA with the data recorded in the system of the service location created, thus saving the additional work for the ticket sales software developers in the preparation of individual requests.

3) Download certificates and check statuses

The completed certificate must be downloaded by the user of the tourist attraction from the NTDSC ATTRAKCIÓ system and sent to the ticket sales software providers.

a. The completed certificates are displayed in the table below the upload block.
b. The certificates are generated immediately, but depending on the load on the system, it can take up to a few minutes, so it's no problem if the requested certificate does not appear immediately.
c. The service location representative can use this table to find out the status of your certificate signing request and certificate, which can be:

   i. Request pending
   ii. Certificate ready
   iii. Error during certificate issuance
   iv. Withdrawal pending
   Withdrawn

d. The service location downloads the certificate from the NTDSC ATTRAKCIÓ system's interface and delivers it to the ticket sales software provider.
e. After this, it needs to be imported to the certificate repository of the TSS software runtime environment (together with the private key, if it was generated elsewhere). It is the responsibility of the TSS manufacturer/service provider to support this step.

In the following cases, a certificate needs to be requested:

- Software migration to another manufacturer's TSS;
- Obtaining a new certificate for an expired certificate, as certificates are valid for two years from the date of issue.
- Certificate replacement request due to corrupted or destroyed private signing keys.

3.5. TSS system configuration in a live environment

The following configuration steps are required to get the TSS software up and running:

- The storage of the TSS software identifier in the NTDSC ATTRAKCIÓ system;
- Store the tax number of the service provider of the service location;
- Store the registration number of the service location;
- Store the NTDSC certificate of a service location.

3.6. Commencement of live data transmission

Once the above configuration settings have been completed, the TSS software is ready for live data transmission.

4. TSS message types, messaging process

4.1. Summary

The TSS interface of the NTDSC ATTRAKCIÓ system is a collection of endpoints serving TSS data communication. Data is received at each data submission endpoint, while data is processed at a common and prioritised verification endpoint.

The TSS interface of the NTDSC ATTRAKCIÓ system distinguishes three endpoints for communication with the TSS software:

- Ticket sales endpoint
- Ticket validation endpoint
- Verification endpoint

TSS software providers should prepare their software products to allow for the possibility to parameterise the frequency of data transmission. The currently specified data sending frequency is 15 minutes.

Receiving endpoints: ticket sales and ticket validation

The NTDSC ATTRAKCIÓ system saves the messages received at the receiving endpoints in a temporary storage and sends a synchronous response to the TSS software about the result of the reception. In this synchronous response, if the reception is successful, the application will reply 200 OK and also send a **processing identifier** to the calling party. In the case of an incorrect request, it returns an error key and an error message, which are described in chapter 5.2 **Receiving a message does not mean that the message has been successfully processed**.

Verification endpoint

The incoming messages are processed asynchronously in the NTDSC ATTRAKCIÓ system. The TSS software can be informed of the result of the processing by periodic polling of the verification endpoint. The request must be accompanied by the **processing identifier** received at the time of submission.

The NTDSC ATTRAKCIÓ system sends a detailed validation result to the requesting party in the response of the verification endpoint, which contains the following elements:

- List of faulty elements, with detailed validation results for each faulty element;
- List of correct items.

The NTDSC ATTRAKCIÓ system is able to process partial data, so there is no need to resubmit correctly returned elements.

If the message has not yet been processed, the message status is RECEIVED. At this point, the processing result cannot be received at the verification endpoint, but can only be obtained by a subsequent, repeated query.

4.2. Order of data submissions

As the data to be sent to the NTDSC ATTRAKCIÓ system are highly interdependent, it is important to note that the order in which data should be sent should always follow the following sequence:

1.) ticket sales data sent to the ticket sales endpoint;
2.) sending ticket validation data at the ticket validation endpoint (if ticket validation is available);
3.) after sending the messages, it is possible to query the processing of the submitted data at the verification endpoint.

In each ticket sales or ticket validation message submitted, it is necessary to link to the service catalogue item associated with the data submission and its modification date to which that submission relates.

The NTDSC ATTRAKCIÓ system expects data submissions on service changes at least daily, prior to the submission of information on ticket sales and ticket validation.

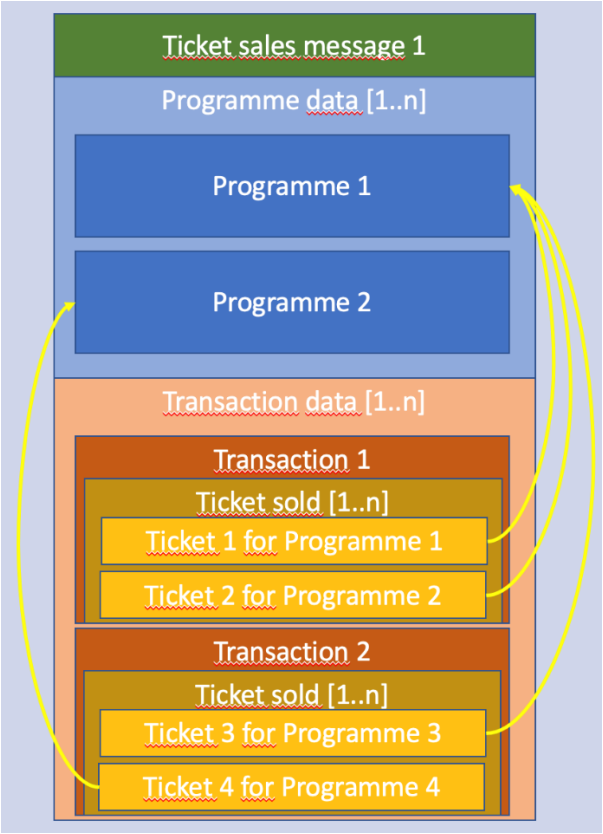
4.3. Submit ticket sales data

The ticket sales information generated since the previous data transmission and the relevant programme data of the sales info sent must be transmitted to the ticket sales data transmission endpoint of the NTDSC ATTRAKCIÓ system. That is, ticket sales messages consist of two larger units:

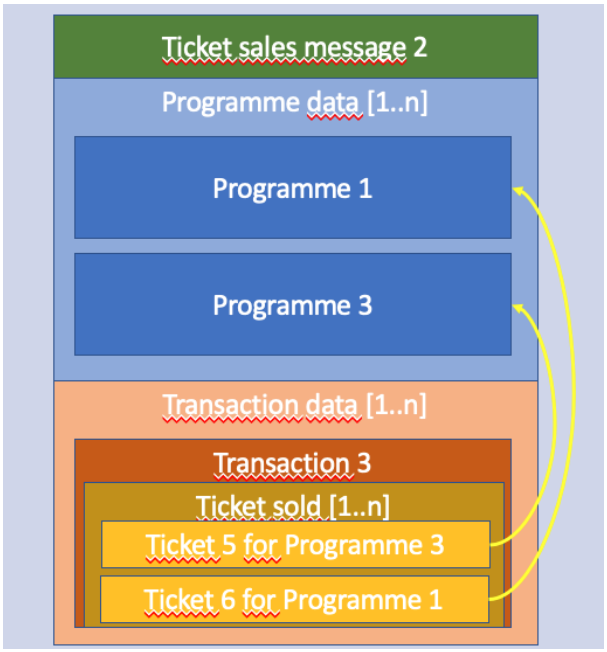
- programme data (service catalogue) and
- sales transactions.

The TSS software must also include the date of the last modification of each program element in the source system. This data is necessary because in the NTDSC ATTRAKCIÓ system, each program element is versioned using the program ID and the date of the last modification. The sending of program elements is mandatory in all cases, even if the data of the program element has not changed since the last data transmission.

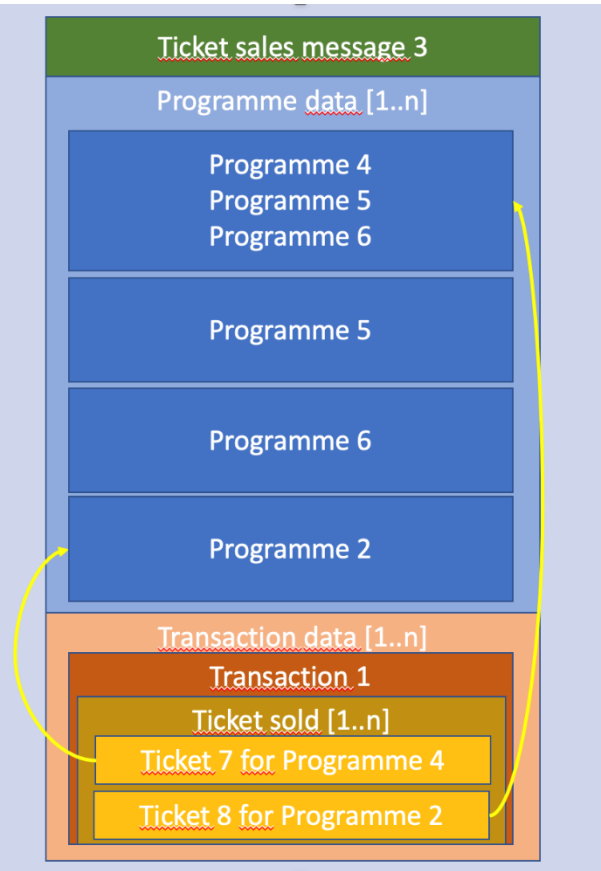
The following diagrams show examples of how ticket sales messages are structured.



This example shows a ticket sales message where the ticket sales software sends data about two transactions. Both transactions involve two or more tickets.
In Transaction 1, both tickets are for Programme 1, while in Transaction 2, one ticket is for Programme 1 and the other for Programme 2.
Since only these two programmes are the only ones for which tickets were sold in the message, the current details of these two programmes must be submitted along with the transaction details.



In the second example, only the data of one transaction with two tickets sold is sent. One for Programme 1, the other for Programme 3. In this case, the details for Programme 2 do not need to be submitted as no tickets have been sold for this programme.
The data for Programme 1 must be submitted regardless of whether there has been a change in its data.
It is important that the values of the fields "Program ID in TSS system" and "Last modification date" sent in the Program data block match the values of the fields with the same name sent in the Sold tickets block.



The third example also shows the data of a Transaction. However, in this example, the first ticket is for a combined programme, which is a combination of the individual programmes offered at the tourist attraction. In this case, in addition to the data of the combined programme, it is also necessary to submit the individual programmes that make up the combined programme, regardless of whether they have not been sold separately. The other ticket is for Programme 2, so its details must also be included in the ticket sales message.

When sending data, the sending software must package the transactions accumulated since the previous data transmission into messages in such a way that the number of messages sent to NTDSC is kept to a minimum. However, in order to ensure successful data transmission, it is essential to ensure that no more than 500 ticket objects are sold in a single ticket sales message. The 500 ticket objects sold can of course be the result of several transactions.If no ticket sales have been made since the previous data provision, no message needs to be sent to the ticket sales endpoint.

When developing integration for the TSS interface, the developer of the TSS system must prepare the application that the value sets of the fields containing ENUMs defined in the interface may change in the future due to adaptation to the changes of market environment.

4.4. Submit ticket validation data

It is necessary to send the ticket validation data transmission to the ticket validation endpoint of the NTDSC ATTRAKCIÓ system for each ticket already sold. If there has been no ticket validation since the previous data provision, it is not necessary to send a message to the ticket validation endpoint.

This is the last transmission that needs to be made, as it is also related to the information sent to the other endpoint.

Ticket validation information - the unique identifier and programme (including last modification date) of the ticket sold for which the ticket used has been validated must be displayed in the ticket validation message. If a ticket is validated more than once (e.g. for multiple admission), data must be sent for each ticket validation.

4.5. Verification of submitted data

For each of the messages sent to the above two data communication endpoints, the calling party receives a unique processing identifier from the NTDSC ATTRAKCIÓ system in a synchronous response. This identifier will be used later to retrieve the result of the processing of the data sent in that particular message.

A beküldött üzenetekre a feldolgozási eredményért történő ellenőrző lekérdezést érdemes NEM rögtön a beküldés után indítani, hogy a fogadó rendszer előbb feldolgozhassa a beküldött üzeneteket és az azonnali bekérdezéssel ne kerüljön felesleges terhelés alá a rendszer. A beküldéstől számított 24 óra belül azonban érdemes végrehajtani az ellenőrző lekérdezést, illetve legkésőbb a beküldéstől számított 1 hónapon belül mindenképp meg kell történnie a lekérdezésnek, mivel utána nem lesz elérhető az információ.

Multiple processing identifiers can be sent to the verification endpoint.

In the response, the result of the data processing is reported broken down by processing identifiers, and within that into successfully and unsuccessfully processed items. In the case of unsuccessful items, the field during the processing of which the error occurred and the value that was sent are returned, and an error key is returned with the related error message.

If the sending party receives a response from the verification endpoint that contains items that have not been processed successfully, the items must be examined, the error corrected and the data resubmitted. Otherwise, the processing of items sent in subsequent data transmissions based on the failed items will also fail. The sending system is responsible for correcting and resubmitting incorrect items.

4.6. Specification description of messages

4.6.1. Structure of message header

All the messages that can be submitted on the interface uniformly have a field group that contains the technical details of the Service Provider, the TSS system and the message.

Field name		JSON name
Service provider data		szolgaltatoAdatok
	Service provider tax number	adoszam
	Service location registration number	szolgHelyRegisztraciosSzam
Message details		uzenetAdatok
	Message send time	uzenetKuldesIdeje
Sending system details		kuldoRendszerAdatok
	TSS system NTDSC system identifier	tssRendszerNTAKAzonosito
	TSS system version number	tssRendszerVerzioszam

4.6.1.1. Service provider data

Type	object
JSON name	szolgaltatoAdatok
Size	-
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Kötelező	Szinkron	NotNull

4.6.1.1.1. Service provider tax number

Type	string
JSON name	adoszam
Size	11
Mandatory	yes
Value set	-
Comment	Tax number of entity submitting the data

Error messages

Validation	Type	Error key

There is a service provider with this tax number but it is deleted	Asynchronous	MissmatchSzolgaltatoAdatokAdoszam
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length = 11	Synchronous	Size
Must be identical in cert and in the field	Synchronous	MissmatchCertificateCnCheck

4.6.1.1.2. Service location registration number

Type	string
JSON name	szolgHelyRegisztraciosSzam
Size	10
Mandatory	yes
Value set	-
Comment	NTDSC registration number of the service location (tourist attraction).

Error messages

Validation	Type	Error key
The service location can be found but the registration number cannot be found	Asynchronous	NotFoundInDbAttrakcio
There is no such service location at all	Asynchronous	NotFoundInDbSzHely
the registration number exists but with a deleted status	Asynchronous	MissingIdSzolgHelyAzonositoSzamAttrakcioban
Mandatory	Synchronous	NotNull
Length = 10	Synchronous	Size
Must be identical in cert and in the field	Synchronous	MissmatchCertificateCnCheck

4.6.1.2. Message details

Type	object
JSON name	uzenetAdatok
Size	-
Mandatory	yes
Value set	-
Comment	-

Hibaüzenetek

Validáció	Típus	Hibakulcs
Kötelező	Szinkron	NotNull

4.6.1.2.1. Message send time

Type	timestamp (RFC 3339 full-time, pl 2022-01-07T19:20:30.45+01:00)
JSON name	uzenetKuldesIdeje
Size	-
Mandatory	yes
Value set	-
Comment	The time of sending the data provision message. Specified by the sender.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Timestamp	Synchronous	MissmatchType
<= sysDate + 2h	Synchronous	Future
>= sysDate - 7d	Synchronous	Past
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.1.3. Sending system details

Type	object
JSON name	kuldoRendszerAdatok
Size	-
Mandatory	yes
Value set	-
Comment	-

Hibaüzenetek

Validáció	Típus	Hibakulcs
Kötelező	Szinkron	NotNull

4.6.1.3.1. TSS system NTDSC system identifier

Type	string
JSON name	tssRendszerNTAKAzonosito
Size	20
Mandatory	yes
Value set	-
Comment	TSS identifier recorded by the HTA in the NTDSC system

Error messages

Validation	Type	Error key
TSS software must be among the validated systems	Asynchronous	MissingTSSSystemInAttrakcio
No TSS software is registered for the service location provided	Asynchronous	ErrorKuldorendszer

Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 20	Synchronous	Size

4.6.1.3.2. TSS system version number

Type	string
JSON name	tssRendszerVerzioszam
Size	20
Mandatory	yes
Value set	-
Comment	Version number of the TSS system concerned

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 20	Synchronous	Size

4.6.2. Ticket sales message structure

The following pages contain the field-level specification of the request message to be sent to the ticket sales endpoint.

Field name			JSON name
Header (see “Structure of message header”)			
Programme data			programAdatok
	Programme ID in the TSS system		tssProgramAzonosito
	Type of programme		programTipusa
	Programme series name		programsorozatNeve
	Program identifier at the service location		szolgHelyProgramAzonosito
	Last modification date		utolsoModositasIdeje
	Programme name		programNeve
	Programme main category		programFokategoria
	Programme subcategory		programAlkategoria
	Programme frequency		programGyakorisaga
	Programme start date		programKezdete
	End of programme, if exists		programVege
	Online programme		onlineProgram
	Location details		helyszinAdatok
		Name of location / tourist attraction	helyszin
		Postal code	iranyitoszam
	Related programmes		kapcsolodoProgramok
		Programme ID in the TSS system	tssProgramAzonosito
		Last modification date	utolsoModositasIdeje
Transaction data			tranzakcioAdatok
	Transaction ID		tranzakcioAzonosito
	Transaction date		tranzakcioIdopontja
	Transaction total sum		tranzakcioVegosszege
	Means of payment		fizetesiModok

		Payment method		fizetesiMod
		Sum paid in HUF		fizetettOsszegHUF
	Sales channel			ertekesitesiCsatorna
	Visitor data			latogatoiAdatok
		Residence of visitor(s)		latogatokLakohelye
		Foreign		kulfoldi
	Tourist card ID			touristCardId
	Combined data provision			osszesitett
	Reason for combined data provision			osszesitettIndoklasa
	Other items			egyebTetelek
		Item identifier		tetelAzonosito
		Item name		tetelMegnevezes
		Category		kategoria
		Quantity		tetelszam
		Gross unit price		bruttoEgysegAr
		VAT category		afaKategoria
	Tickets sold			eladottJegyek
		Ticket identifier		jegyAzonosito
		Ticket name		jegyMegnevezes
		Gross Price		bruttoAr
		VAT category		afaKategoria
		Tourist card ID		touristCardId
		Max number of entries		maxBelepesekSzama
		Number of persons		szemelyekSzama
		Start of validity		ervenyessegKezdete
		Transport service provider data		kozlekedesiAdatok
			Place of departure (ZIP code)	indulasiHelyPir
			Place of departure (coord)	indulasiHelyCoord
			Departure time	indulasiIdopont
			Place of arrival (ZIP code)	erkezesiHelyPir
			Place of arrival (coord)	erkezesiHelyCoord
			Arrival time	erkezesiIdopont
			Return	retur
		NTDSC system category		ntakRendszerKategoria
		Ticket validity type		jegyErvenyessegTipusa
		Discounts		kedvezmenyek
		Age group		korcsoport
		Programme identifier		programazonosito
			Programme ID within TSS	tssProgramAzonosito
			Last modification date	utolsoModositasIdeje
		Used immediately?		azonnalFelhasznalt

4.6.2.1. Programme data

Type	array
JSON name	programAdatok
Size	1..n
Mandatory	yes
Value set	-
Comment	-

Hibaüzenetek

Validáció	Típus	Hibakulcs
Kötelező	Szinkron	NotNull

4.6.2.1.1. Programme ID in the TSS system

Type	string
JSON name	tssProgramAzonosito
Size	50
Mandatory	yes
Value set	-
Comment	Unique identifier of the programme in the tourist attraction's TSS system (searchable in NTDSC: progazon.+szolghelyazon+tssrendszerazon+utolsomodositas)

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 50	Synchronous	Size

4.6.2.1.2. Type of programme

Type	enum
JSON name	programTipusa
Size	-
Mandatory	yes
Value set	ONALLO_PROGRAM PROGRAMSOROZAT_RESZE KOMBINALT_PROGRAM
Comment	Combined programme: A programme of own programmes For stand-alone programmes, title information is mandatory and there is no associated programme block (ha szerepel érték ebben a mezőben, Conflict hibakulcsot ad vissza az interfész) and programme series name field In the case of combined programmes, no location details are provided, but the related programme details are required. Part of a series of programmes: At this time, a combined program cannot be part of a series of programs. is interpreted as a stand-alone program, the data to be entered is the same as for the stand-alone program, plus the programme series name field.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict

4.6.2.1.3. Programme series name

Type	string
JSON name	programsorozatNeve
Size	200
Mandatory	conditional
Value set	-
Comment	Mandatory if Programme type is PROGRAMSOROZAT_RESZE

Error messages

Validation	Type	Error key
Length <= 200	Synchronous	Size
Mandatory	Synchronous	NotNull

4.6.2.1.4. Program identifier at the service location

Type	string
JSON name	szolgHelyProgramAzonosito
Size	50
Mandatory	no
Value set	-
Comment	This field is required if more than one ticket sales software is in use at the same tourist attraction, therefore selling tickets for a given programme is possible from more than one software. Given that usually the internal identifiers used by each software cannot be matched, this field has been created to indicate in the data submitted that sales from two different TSS software were made to the same programme. TSS software developers are required to provide the option on the user interface for manually setting this field so that the users can specify the program identifier, making sure it is consistent between applications. This way, when creating statistics, the number and sum of ticket sales can be aggregated by programme, even if they were made in more than one software.

Error messages

Validation	Type	Error key
Length <= 50	Synchronous	Size
empty string (“”)	Synchronous	NotEmpty

4.6.2.1.5. Last modification date

Type	timestamp
JSON name	utolsoModositasIdeje
Size	-
Mandatory	yes
Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00 Ha a dátum mező 1800-nál korábbi évszámot tartalmaz, az interfész Conflict hibaüzenetet ad vissza.

Error messages

Validation	Type	Error key
Timestamp	Synchronous	MissmatchType
<= sysDate	Synchronous	Future
Mandatory	Synchronous	NotNull
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.2.1.6. Programme name

Type	string
JSON name	programNeve
Size	100
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 100	Synchronous	Size

4.6.2.1.7. Programme main category

Type	enum
JSON name	programFokategoria
Size	-
Mandatory	yes
Value set	see Programme categories annex
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.1.8. Programme subcategory

Type	enum
JSON name	programAlkategoria
Size	-
Mandatory	yes
Value set	see Programme categories annex
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum
The value must be present in a list filtered by main category	Synchronous	Conflict

4.6.2.1.9. Programme frequency

Type	enum
JSON name	programGyakorisaga
Size	-
Mandatory	yes
Value set	ALLANDO EGYSZERI NEHANY_ALKALMAS IDOSZAKOS RENDSZERESEN_ISMETLODO
Comment	Some suitable categories indicate programmes that are organized multiple times, but independently on occasion.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.1.10. Programme start date

Type	timestamp
JSON name	programKezdete
Size	-
Mandatory	yes

Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00 Ha a dátum mező 1800-nál korábbi évszámot tartalmaz, az interfész Conflict hibaüzenetet ad vissza.

Error messages

Validation	Type	Error key
Timestamp	Synchronous	MissmatchType
Mandatory	Synchronous	NotNull
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.2.1.11. End of programme, if exists

Type	timestamp
JSON name	programVege
Size	-
Mandatory	no
Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00

Error messages

Validation	Type	Error key
Timestamp	Synchronous	MissmatchType
>= programKezdete	Synchronous	Past
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.2.1.12. Online programme

Type	bool
JSON name	onlineProgram
Size	-
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
bool	Synchronous	MissmatchType

4.6.2.1.13. Location details

Type	object

JSON name	helyszinAdatok
Size	-
Mandatory	conditional
Value set	-
Comment	Not required for combined or online programmes, required field for normal offline programmes

4.6.2.1.13.1. Name of location / tourist attraction

Type	string
JSON name	helyszin
Size	200
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 200	Synchronous	Size

4.6.2.1.13.2. Postal code

Type	int
JSON name	iranyitoszam
Size	4
Mandatory	yes
Value set	-
Comment	Postal address database value

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Length = 4	Synchronous	Size
Integer	Synchronous	MissmatchType
Included in postal address database	Synchronous	NotFoundInDb

4.6.2.1.14. Related programmes

Type	array
JSON name	kapcsolodoProgramok
Size	2..n
Mandatory	conditional
Value set	-
Comment	Sending this block is mandatory if it is a combined programme. In this case, the identifiers of the other

	programmes of the tourist attraction are given here. In this case, sending the Location block is not by default.

4.6.2.1.14.1. Programme ID within TSS

Type	string
JSON name	tssProgramAzonosito
Size	50
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Programme cannot be found	Synchronous	MissingProgram
Circular reference !!!	Synchronous	SelfReference

4.6.2.1.14.2. Last modification date

Type	timestamp
JSON name	utolsoModositasIdeje
Size	-
Mandatory	yes
Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00 Ha a dátum mező 1800-nál korábbi évszámot tartalmaz, az interfész Conflict hibaüzenetet ad vissza.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Timestamp	Synchronous	MissmatchType
<= sysDate	Synchronous	Future
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.2.2. Transaction data

Type	array
JSON name	tranzakcioAdatok
Size	1..n
Mandatory	yes

Value set	-
Comment	-

Hibaüzenetek

Validáció	Típus	Hibakulcs
Kötelező	Szinkron	NotNull

4.6.2.2.1. Transaction ID

Type	string
JSON name	tranzakcioAzonosito
Size	100
Mandatory	yes
Value set	-
Comment	Unique identifier of the transaction in the tourist attraction's Ticket Sales System

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 100	Synchronous	Size
Unique (servlocationregnumber + software identifier + tr.identifier)	Asynchronous	UniqueConstraint

4.6.2.2.2. Transaction date

Type	timestamp
JSON name	tranzakcioIdopontja
Size	-
Mandatory	yes
Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00 Ha a dátum mező 1800-nál korábbi évszámot tartalmaz, az interfész Conflict hibaüzenetet ad vissza.

Error messages

Validation	Type	Error key
Timestamp	Synchronous	MissmatchType
<= sysDate	Synchronous	Future
Mandatory	Synchronous	NotNull
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.2.2.3. Transaction total sum

Type	int
JSON name	tranzakcioVegosszege
Size	-
Mandatory	yes
Value set	-
Comment	The sum of the sum of the gross price multiplied by the amount of other items and the sum of the gross price of tickets sold The total sum of the transaction is the total gross price of the transaction items [*∑(other item's amount * gross unit price) + ∑(gross price of tickets sold)]* rounded to an integer (rounding by merchant).

Error messages

Validation	Type	Error key
Sum of payment methods	Synchronous	Conflict
Sum of tickets and other items	Synchronous	Conflict
Mandatory	Synchronous	NotNull
integer	Synchronous	MissmatchType

4.6.2.2.4. Means of payment

Type	array
JSON name	fizetesiModok
Size	1..n
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.4.1. Payment method

Type	enum
JSON name	fizetesiMod
Size	-
Mandatory	yes
Value set	KESZPENZHUF KESZPENZEUR SZEPKARTYA BANKKARTYA ATUTALAS EGYEB VOUCHER SZOBAHITEL

	KEREKITES
Comment	List only the payment method that were used in the given transaction. Calculation method of rounding value: paid sum - payable sum (i.e., if payable sum is HUF 54, the payment methods: KP:55; KER:-1)

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.4.2. Sum paid in HUF

Type	int
JSON name	fizetettOsszegHUF
Size	-
Mandatory	yes
Value set	-
Comment	Amount paid by in HUF

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
integer	Synchronous	MissmatchType
In the case of KESZPENZHUF 0 or 5 ending is mandatory	Synchronous	Conflict

4.6.2.2.5. Sales channel

Type	enum
JSON name	ertekesitesiCsatorna
Size	-
Mandatory	yes
Value set	ONLINE AUTOMATA HELYSZINI UTAZASI_IRODAN_KERESZTULI
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.6. Visitor data

Type	object
JSON name	latogatoiAdatok
Size	-
Mandatory	yes
Value set	-
Comment	-

Hibaüzenetek

Validáció	Típus	Hibakulcs
Kötelező	Szinkron	NotNull

4.6.2.2.6.1. Residence of visitor(s)

Type	string
JSON name	latogatokLakohelye
Size	4
Mandatory	yes
Value set	-
Comment	Postal code or ISO2 country code or ""

Error messages

Validation	Type	Error key
existing zip code or country code or empty string	Synchronous	NotFoundInDb
Mandatory	Synchronous	NotNull
Length <= 4	Synchronous	Size

4.6.2.2.6.2. Foreign

Type	bool
JSON name	kulfoldi
Size	-
Mandatory	yes
Value set	-
Comment	To be filled in all cases.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Bool	Synchronous	MissmatchType

4.6.2.2.7. Tourist card ID

Type	string
JSON name	touristCardId
Size	30
Mandatory	no
Value set	-
Comment	Tourist Card ID

Error messages

Validation	Type	Error key
Length <= 30	Synchronous	Size

4.6.2.2.8. Combined data provision

Type	bool
JSON name	osszesitett
Size	-
Mandatory	yes
Value set	-
Comment	It indicates ticket sales messages that contain the combined sales of a longer period (max. 1 business day), or a special aggregated period. It can be used in case of a service outage, (e.g. power outage, system downtime), or in accordance with certain business logics like monthly aggregates (of guests arriving from partner hotels, or treatments subsidized by OEP/NEAK – the National Health Insurance Fund of Hungary). It is to be sent with false data in the case of normal data provision

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Bool	Synchronous	MissmatchType

4.6.2.2.9. Reason for combined data provision

Type	enum
JSON name	osszesitettIndoklasa
Size	-
Mandatory	conditional
Value set	UZEMSZUNET_ARAMSZOLGALTATAS_TERVEZETT_KIMARADASA_MIATT UZEMSZUNET_INTERNETSZOLGALTATAS_TERVEZETT_KIMARADASA_MIATT UZEMZAVAR_ARAMSZOLGALTATAS_ATMENETI_KIMARADASA_MIATT UZEMZAVAR_INTERNETSZOLGALTATAS_ATMENETI_KIMARADASA_MIATT UZEMZAVAR_TERMESZETI_KATASZTROFA_MIATT

	UZEMZAVAR_LEJART_TANUSITVANY_MIATT UZEMZAVAR_MUSZAKI_HIBA_MIATT HAVI_OSSZESITES_PARTNER_SZALLODA HAVI_OSSZESITES_OEP
Comment	Mandatory if osszesitett = true A 'HAVI_OSSZESITES_PARTNER_SZALLODA' elsősorban közfürdők esetében használatos, amikor havonta egyszer összesítve jelentik azon vendégek számát, akik partner szállodákból érkeztek. A funkció bármely attrakció számára elérhető, ahol releváns az együttműködés nyomon követése. A 'HAVI_OSSZESITES_OEP' segíti a támogatott kezelések forgalmának elemzését. Elsősorban közfürdők esetében használatos, amikor havonta egyszer összesítve jelentik azon látogatók számát, akik az Országos Egészségbiztosítási Pénztár (OEP) (jogutódja a Nemzeti Egészségbiztosítási Alapkezelő (NEAK)) által támogatott, egészségügyi kezelés miatt vették igénybe a szolgáltatást.

Error messages

Validation	Type	Error key
mandatory if osszesitett = true	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.10. Other items

Type	array
JSON name	egyebTetelek
Size	1..n
Mandatory	no
Value set	-
Comment	The block is not mandatory if it has no elements, if it contains elements then the element fields must be filled in

4.6.2.2.10.1. Item identifier

Type	string
JSON name	tetelAzonosito
Size	100
Mandatory	yes
Value set	-
Comment	Unique identifier of the item in the tourist attraction's Ticket Sales System

Error messages

Validation	Type	Error key
Mandatory if the item can be found in the block	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 100	Synchronous	Size

Unique (servlocationregnumber + software identifier + item identifier)	Asynchronous	UniqueConstraint

4.6.2.2.10.2. Item name

Type	string
JSON name	tetelMegnevezes
Size	50
Mandatory	yes
Value set	-
Comment	Item name

Error messages

Validation	Type	Error key
Mandatory if the item can be found in the block	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 50	Synchronous	Size

4.6.2.2.10.3. Category

Type	enum
JSON name	kategoria
Size	-
Mandatory	yes
Value set	TERMEK SZOLGALTATAS ETEL ITAL KEDVEZMENY
Comment	Item category, discount rate in a breakdown of VAT categories, in HUF, as negative integers

Error messages

Validation	Type	Error key
Mandatory if the item can be found in the block	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.10.4. Quantity

Type	double
JSON name	tetelszam
Size	-
Mandatory	yes
Value set	-
Comment	Number of items sold

Error messages

Validation	Type	Error key
Mandatory if the item can be found in the block	Synchronous	NotNull
Double	Synchronous	MissmatchType
> 0	Synchronous	Negative
<> 0	Synchronous	Zero

4.6.2.2.10.5. Gross unit price

Type	double
JSON name	bruttoEgysegAr
Size	-
Mandatory	yes
Value set	-
Comment	Gross unit price of item

Error messages

Validation	Type	Error key
Mandatory if the item can be found in the block	Synchronous	NotNull
double	Synchronous	MissmatchType

4.6.2.2.10.6. VAT category

Type	enum
JSON name	afaKategoria
Size	-
Mandatory	yes
Value set	A_5 B_18 C_27

	D_AJT E_0
Comment	-

Error messages

Validation	Type	Error key
Mandatory if the item can be found in the block	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.11. Tickets sold

Type	array
JSON name	eladottJegyek
Size	1..n
Mandatory	yes
Value set	-
Comment	-

Hibaüzenetek

Validáció	Típus	Hibakulcs
Kötelező	Szinkron	NotNull

4.6.2.2.11.1. Ticket identifier

Type	string
JSON name	jegyAzonosito
Size	100
Mandatory	yes
Value set	-
Comment	The unique identifier of the ticket in the tourist attraction's Ticket Sales System

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 100	Synchronous	Size
Unique (servlocationregnumber + software identifier + ticket identifier)	Asynchronous	UniqueConstraint

4.6.2.2.11.2. Ticket name

Type	string
JSON name	jegyMegnevezes

Size	50
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 50	Synchronous	Size

4.6.2.2.11.3. Gross Price

Type	double
JSON name	bruttoAr
Size	-
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
> 0	Synchronous	Negative
Integer	Synchronous	MissmatchType

4.6.2.2.11.4. VAT category

Type	enum
JSON name	afaKategoria
Size	-
Mandatory	yes
Value set	A_5 B_18 C_27 D_AJT E_0
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.11.5. Tourist card ID

Type	string
JSON name	touristCardId
Size	30
Mandatory	no
Value set	-
Comment	Tourist Card ID

Error messages

Validation	Type	Error key
Length <= 30	Synchronous	Size

4.6.2.2.11.6. Max number of entries

Type	int
JSON name	maxBelepesekSzama
Size	-
Mandatory	yes
Value set	-
Comment	If the ticket is not a stand-alone ticket (e.g. audio guide), it is set to 0. For unlimited access, the value is: 999

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Integer	Synchronous	MissmatchType

4.6.2.2.11.7. Number of persons

Type	int
JSON name	szemelyekSzama
Size	-
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Integer	Synchronous	MissmatchType

4.6.2.2.11.8. Start of validity

Type	timestamp
JSON name	ervenyessegKezdete
Size	-
Mandatory	yes
Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00

	Ha a dátum mező 1800-nál korábbi évszámot tartalmaz, az interfész Conflict hibaüzenetet ad vissza.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.2.2.11.9. Transport service provider data

Type	object
JSON name	kozlekedesiAdatok
Size	-
Mandatory	no
Value set	-
Comment	Block to be filled only for BAHART and MAHART service providers

4.6.2.2.11.9.1. Place of departure (ZIP code)

Type	string
JSON name	indulasiHelyPir
Size	4
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Included in postal address database	Synchronous	NotFoundInDb
Length <= 4	Synchronous	Size

4.6.2.2.11.9.2. Place of departure (coord)

Type	string
JSON name	indulasiHelyCoord
Size	20
Mandatory	yes
Value set	-
Comment	Geographical coordinates in decimal degree format, to 6 decimal places, separated by commas

Error messages

Validation	Type	Error key
Length <= 20	Synchronous	Size

4.6.2.2.11.9.3. Departure time

Type	timestamp
JSON name	indulasiIdopont
Size	-
Mandatory	yes
Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00 Ha a dátum mező 1800-nál korábbi évszámot tartalmaz, az interfész Conflict hibaüzenetet ad vissza.

Error messages

Validation	Type	Error key
Timestamp	Synchronous	MissmatchType
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.2.2.11.9.4. Place of arrival (ZIP code)

Type	string
JSON name	erkezesiHelyPir
Size	4
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Included in postal address database	Synchronous	NotFoundInDb
Length <= 4	Synchronous	Size

4.6.2.2.11.9.5. Place of arrival (coord)

Type	string
JSON name	erkezesiHelyCoord
Size	20
Mandatory	yes
Value set	-
Comment	Geographical coordinates in decimal degree format, to 6 decimal places, separated by commas

Error messages

Validation	Type	Error key

Length <= 20	Synchronous	Size

4.6.2.2.11.9.6. Arrival time

Type	timestamp
JSON name	erkezesIdopont
Size	-
Mandatory	yes
Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00

Error messages

Validation	Type	Error key
Timestamp	Synchronous	MissmatchType
>= indulasIdopont	Synchronous	Past
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.2.2.11.9.7. Return

Type	bool
JSON name	retur
Size	-
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
bool	Synchronous	MissmatchType

4.6.2.2.11.10. NTDSC system category

Type	enum
JSON name	ntakRendszerKategoria
Size	-
Mandatory	yes
Value set	ONALLO_BELEPESRE_NEM_JOGOSITO EGYENI CSALADI CSOPORTOS
Comment	

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.11.11. Ticket validity type

Type	enum
JSON name	jegyErvenyessegTipusa
Size	-
Mandatory	yes
Value set	EGYEB HETI_JEGY HAVI_JEGY EGYSZERI NAPI_JEGY NEGYEDEVES FELEVES EVES
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.11.12. Discounts

Type	enum
JSON name	kedvezmeny
Size	1..n
Mandatory	yes
Value set	CSOPORTOS_KEDVEZMENY CSALADI_KEDVEZMENY EGYEB_KARTYAKEDVEZMENYEK EGYEB_KEDVEZMENYEK DIAK_KEDVEZMENY FOGYATEKKAL_ELOK_KEDVEZMENYE HELYI_LAKOS_KEDVEZMENY GYEREKKEDVEZMENY NYUGDIJAS_KEDVEZMENY SZEPKORU_KEDVEZMENY SZERZODESES_PARTNERI_KEDVEZMENYEK SZAKMAI_SZERVEZETEK_KEDVEZMENYE TARSINTEZMENYI_KEDVEZMENYEK TURISZTIKAI_KARTYAKEDVEZMENY UTAZASI_IRODAI_KEDVEZMENY
Comment	If the ticket includes a discount or discounts, a list of them.

Error messages

Validation	Type	Error key
Enum	Synchronous	ErrorEnum

4.6.2.2.11.13. Age group

Type	enum
JSON name	korcsoport
Size	-
Mandatory	yes
Value set	VEGYES NYUGDIJAS IDOS GYEREK FELNOTT DIAK
Comment	Recommended age limits child: 0-6 youth: 7-25 adult: 26-63 pensioner: 64-69 senior: 70- mixed: 0-99

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.2.2.11.14. Programme identifier

Type	object
JSON name	programazonosito
Size	-
Mandatory	yes
Value set	-
Comment	-

4.6.2.2.11.14.1. Programme ID within TSS

Type	string
JSON name	tssProgramAzonosito
Size	50
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty

Programme cannot be found (id+ver)	Synchronous	MissingProgram

4.6.2.2.11.14.2. Last modification date

Type	timestamp
JSON name	utolsoModositasIdeje
Size	-
Mandatory	yes
Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00 Ha a dátum mező 1800-nál korábbi évszámot tartalmaz, az interfész Conflict hibaüzenetet ad vissza.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Timestamp	Synchronous	MissmatchType
<= sysDate	Synchronous	Future
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.2.2.11.15. Used immediately?

Type	bool
JSON name	azonnalFelhasznalt
Size	-
Mandatory	no
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Bool	Synchronous	MissmatchType

4.6.3. Ticket validation message structure

Field name		JSON name
Header (see “Structure of message header”)		
Sales system details		ertekesitoRendszerAdatok
	Sales TSS system identifier in NTDSC system	ertekesitoRendszerNTAKAzonosito
	Sales TSS system version number	ertekesitoRendszerVerzioszam
	Sales Service location registration number	ertekesitoSzolgHelyRegSzam
Tickets		jegyek
	Ticket identifier	jegyAzonosito

	Validation identifier		ervenyesitesiAzonosito
	How to use		felhasznalasModja
	Number of persons admitted		belepettSzemelyekSzama
	Date of validation		ervenyesitesIdopontja
	Programme identifier		programazonosito
		Programme ID in the TSS system	tssProgramAzonosito
		Last modification date	utolsoModositasIdeje

4.6.3.1. Sales system details

Type	object
JSON name	ertekesitoRendszerAdatok
Size	
Mandatory	no
Value set	-
Comment	-

4.6.3.1.1. Sales TSS system identifier in NTDSC system

Type	string
JSON name	ertekesitoRendszerNTAKAzonosito
Size	20
Mandatory	yes
Value set	-
Comment	Sales TSS identification number recorded by the HTA in the NTDSC system

Error messages

Validation	Type	Error key
	Asynchronous	MissingTSSSystemInAttrakcio
The TSS software provided is not registered for the service location	Asynchronous	ErrorKuldorendszer
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 20	Synchronous	Size

4.6.3.1.1. Sales TSS system version number

Type	string
JSON name	ertekesitoRendszerVerzioszam
Size	20
Mandatory	yes
Value set	-
Comment	Version number of TSS system performing the sale

Error messages

Validation	Type	Error key

Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Length <= 20	Synchronous	Size

4.6.3.1.1. Sales Service location registration number

Type	string
JSON name	ertekesitoSzolgHelyRegSzam
Size	10
Mandatory	yes
Value set	-
Comment	NTDSC registration number of the service location (tourist attraction) performing the sale.

Error messages

Validation	Type	Error key
The service location can be found but the registration number cannot be found.	Asynchronous	NotFoundInDbAttrakcio
There is no such service location at all.	Asynchronous	NotFoundInDbSzHely
There is a registration number but it is deleted	Asynchronous	MissingIdSzolgHelyAzonositoSzamAttrakcioban
Mandatory	Synchronous	NotNull
Length = 10	Synchronous	Size
Must be identical in cert and in the field.	Synchronous	MissmatchCertificateCnCheck

4.6.3.1. Tickets

Type	array
JSON name	tranzakcioAdatok
Size	1..n
Mandatory	yes
Value set	-
Comment	-

4.6.3.1.1. Ticket identifier

Type	string
JSON name	jegyAzonosito
Size	100
Mandatory	yes
Value set	-
Comment	Unique identifier of the ticket in the tourist attraction's Ticket Sales System

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
	Asynchronous	NotFoundInDb
length <= 100	Synchronous	Size

4.6.3.1.1. Validation identifier

Type	string
JSON name	ervenyesitesiAzonosito
Size	100
Mandatory	yes
Value set	-

Comment	Unique identifier of the entry in the tourist attraction's Ticket Sales System

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty
Unique (servlocationregnumber + software identifier + validation identifier)	Asynchronous	UniqueConstraint
length <= 100	Synchronous	Size

4.6.3.1.1. How to use

Type	enum
JSON name	felhasznalasModja
Size	-
Mandatory	yes
Value set	TORLES BELEPES KILEPES
Comment	How to use the ticket

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Enum	Synchronous	ErrorEnum

4.6.3.1.1. Number of persons admitted

Type	int
JSON name	belepettSzemelyekSzama
Size	-
Mandatory	yes
Value set	-
Comment	If the ticket is not a stand-alone ticket, 0 should be sent.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull

4.6.3.1.1. Date of validation

Type	timestamp
JSON name	ervenyesitesIdopontja
Size	-
Mandatory	yes
Value set	-

Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00 Ha a dátum mező 1800-nál korábbi évszámot tartalmaz, az interfész Conflict hibaüzenetet ad vissza.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.3.1.1. Programme identifier

Type	object
JSON name	programazonosito
Size	-
Mandatory	yes
Value set	-
Comment	-

4.6.3.1.1.1. Programme ID in the TSS system

Type	string
JSON name	tssProgramAzonosito
Size	50
Mandatory	yes
Value set	-
Comment	-

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
Mandatory	Synchronous	NotEmpty

4.6.3.1.1.2. Last modification date

Type	timestamp
JSON name	utolsoModositasIdeje
Size	-
Mandatory	yes
Value set	-
Comment	RFC 3339 standard full-time, e.g. 2022-01-07T19:20:30.45+01:00 Ha a dátum mező 1800-nál korábbi évszámot tartalmaz, az interfész Conflict hibaüzenetet ad vissza.

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
A mező értéke ellentmondásban áll a függőségeivel!	Szinkron	Conflict
Nem lehet negatív a mező értéke!	Szinkron	Negative

4.6.4. Verification request message structure

Field name		JSON name
Header (see “Structure of message header”)		
NTDSC system unique message identifiers		feldolgozasAzonositok
	NTDSC system unique message identifier	feldolgozasAzonosito

4.6.4.1. NTDSC system unique message identifiers

Type	array
JSON name	feldolgozasAzonositok
Size	-
Mandatory	yes
Value set	-
Comment	-

4.6.4.1.1. NTDSC system unique message identifier

Type	uuid
JSON name	feldolgozasAzonosito
Size	-
Mandatory	yes
Value set	-
Comment	Unique processing identifier generated by the TSS IF system in the response to messages sent to the other 2 endpoints

Error messages

Validation	Type	Error key
Mandatory	Synchronous	NotNull
uuid	Synchronous	MissmatchType

4.6.5. Response messages

4.6.5.1. In case of acceptable requests

4.6.5.1.1. In case of Order summary or Daily closure message

If the data in the submitted ticket sales or the ticket validation message are correct based on preliminary checks and the system can accept the message, then a processing identifier is sent in the synchronous response with which the status of the message received can be queried at the verification endpoint.

Field name	JSON name	Type
NTDSC system unique processing identifier	feldolgozasAzonosito	UUID

4.6.5.1.2. In case of a verification endpoint message

In case of a successful request message to the verification endpoint, a message is sent with the following structure. The response message contains in every case the Processing identifiers queried and the related Status.

If the message belonging to the Processing identifier has not been processed yet, the Status field value will be „RECEIVED” and the other blocks containing successful/unsuccessful items will not appear.

If the message belonging to the Processing identifier has been processed, its content will be listed by item (ticket sales or ticket validation) in blocks of successful and unsuccessful items. The values of the type field can be JEGYERTEKESITES, or JEGYERVENYESITES. In the identifier field sent by TSS, in case of ticket sales, the transaction identifier, and in case of ticket validation, the validation identifier are returned. In the case of successful items the error that occurred will also be detailed. The status field values can be the following: BEFOGADVA, TELJESEN_HIBAS, RESZBEN_SIKERES, TELJESEN_SIKERES, UJRA_KULDENDO,

If the message belonging to the Processing identifier cannot be processed due to an error of the message header, it will be detailed in the Header errors block. In such cases, the blocks containing successful/unsuccessful messages will not appear.

Field name	JSON name
NTDSC system unique message identifiers	uzenetValaszok
NTDSC system unique message identifier	feldolgozasAzonosito
Header errors	fejlecHibak
Field name	mezoNeve
Value sent	kuldottErtek
Error key	hibaKulcs
Error message	hibaUzenet
Items processed successfully	sikeresUzenetek
Type	tipus
Identifier sent by TSS	tssAzonosito
Items processed unsuccessfully	sikertelenUzenetek
Type	tipus
Identifier sent by TSS	tssAzonosito
List of errors	uzenetHibak
Field name	mezoNeve
Value sent	kuldottErtek
Error key	hibaKulcs
Error message	hibaUzenet

	Status	statusz

4.6.5.2. In the case of unacceptable requests

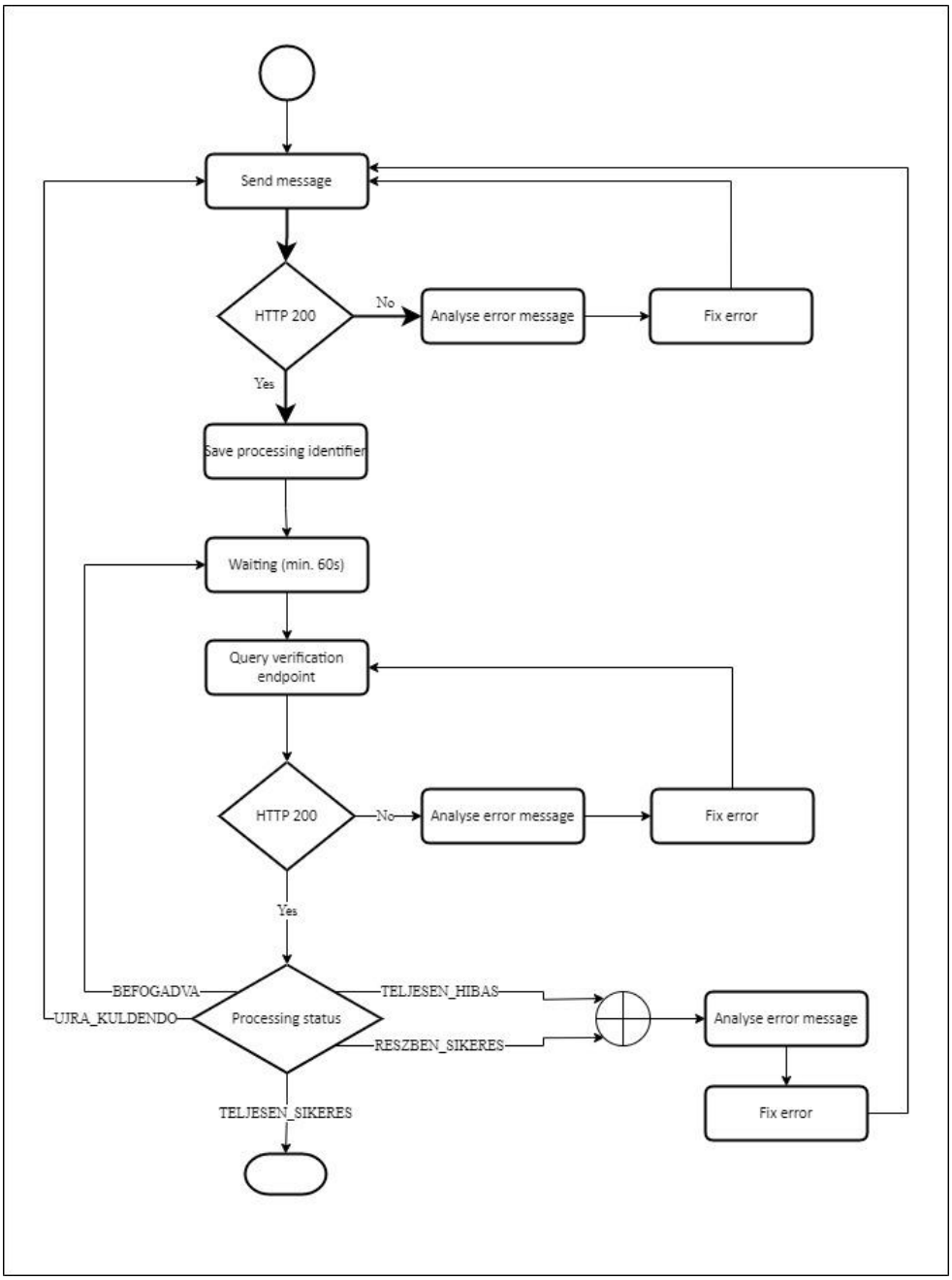
If the message submitted, based on the information it contains, is incorrect and the system cannot accept it, the location and reason of the error will be sent in a synchronous response in the following message structure.

Field name		JSON name	Type
List of errors		uzenetHibak	array
	Field name	mezoNeve	string
	Value sent	kuldottErtek	string
	Error key	hibaKulcs	string
	Error message	hibaUzenet	string

5. Formal requirements

This chapter details the formal requirements, technical information and error keys related to messaging.

5.1. General data submission requirements



Ticket sales and ticket validation data can be submitted to the NTDSC ATTRAKCIÓ system via HTTP REST call endpoints. To ensure the integrity of data submissions, the submitted data must be signed with a digital signature. For the submitted data, the NTDSC ATTRAKCIÓ system first checks the certificate and then the

digital signature. If there is any problem with these, it will send a synchronous error response and will not commence processing the submitted data.

If both the certificate and the digital signature are valid, the system performs a syntactic check on the data sent in the HTTP REST body, i.e. it checks whether all the mandatory data is present in the request. These checks are described in detail in the document referred to in chapter 5.2.

If the submission passes the NTDSC ATTRAKCIÓ system's syntax check, a processing identifier is returned for the submitted request. This means that the NTDSC ATTRAKCIÓ system has accepted the request for processing, in which case the processing status will be RECEIVED.

The sending TSS system can use the processing identifier to retrieve the current result of the status of the submitted data. The result of the processing of the submitted data can be queried at the verification endpoint (/api/ellenorzes/statusz). Sending TSS systems can use this verification endpoint to retrieve the status of successfully and/or unsuccessfully processed data. The status of more than one submission (processing identifier) can be queried through the verification endpoint at one time. If the data submission fails, the list of errors contains the values to be corrected. In such a case, the errors in the list of errors shall be corrected and the corrected data shall be resubmitted to the NTDSC ATTRAKCIÓ system. Only the unsuccessfully processed data needs to be resent to the sending TSS system.

After receiving, the requests (the status of the submitted request is set to RECEIVED at the verification endpoint) are further processed in the NTDSC ATTRAKCIÓ system. Here the system re-verifies the certificate and digital signature sent. Both the certificate and the digital signature must be sent as a key-value pair in the HTTP header of the REST query. The certificate is the x-certificate key, and its value is the BASE64-encoded certificate. The x-jws-signature is the key, and its value will be the generated key.

Example:

x-certificate :

LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUhEVENDQXZXZ0F3SUJBZ0lDRUFJd0RRWUpLb1pJaHZjTkFRRUxCUUF3Z1k4e
ER6QU5CZ05WQkFNTUJrRlQKUkRFeU16RU5NQXNHQTFVRUNBd0VVR1Z6ZERFTE1Ba0dBMVVFQmhNQ1NIVXhJekFoQmdrcW
hraUc5dzBCQ1FFVwpGR2x1Wm05QWFXNTBaWEp0WldScFlYUmxMbWgxTVNFd0h3WURWUVFLREJoSmJuUmxjbTFsWkdsaG
RHVWdTMkZzCmJXRnVJRXRtZEM0eEdEQVdCZ05WQkFzTUQwbFVJR2x1ZEdWeWJXVmthV0YwWlRBZUZ3MHlNREExTWpBeE5
EUTIKTVRoYUZ3MHlNVEExTWpBeE5EUTJNVGhhTURjeEVUQVBCZ05WQkFNTUNFaElVa3BSTVV0Q01RMHdDd1lEVlFRSQpEQVJ
RWlhOME1STXdFUVlEVlFRS0RBcExZV3h0WVc0Z1QzSm5NSUlCSWpBTkJna3Foa2lHOXcwQkFRUZBQU9DCkFROEFNSUlCQ2dL
Q0FRRUF5QlZtQlN4akcwWjdCakNHR2wzMjQ5aEV5Nnloa0IxMUdkL214MlQ4Wnh4NmZwbGQKTmt5WVhVR0hsZklKV2p3YkJy
V3NxNVUvWEpLM2p1SG5lTytCQjFkZThYRlJNSWwyMkJRSnVOWE9RRTU4dUNpagovZHhHc3k4WlpNM2hHMWxCSCtva1A0NH
M5Q095R0F6WmVUaXgrNVhUcHc2MjBDU0VjODl2TWwvNlNhSnBVcHRUCk5XclNJYVEwSXN4M2Y2MWhBRzl6OVFybkF4dEJIa
nZrY2FVVW9RL0w0Y0dsaEh5QzF1c0ZvVS8xWUdlMUVUSzQKempNN0lza013ekhweFQ2ZDZGaitlUzV5MlVaQjhkR0ZGTjZMS25
hV2ZjNFN3YWtrRUdXSW5yTDhxREt4b1JUVAp0ajc1T2hBSzlzOHY5eUlqMCtxMkhBeGdReFF3TXJvMFZQR21od0lEQVFBQm80S
EpNSUhHTUF3R0ExVWRFd0VCCi93UUNNQUF3SFFZRFZSME9CQllFRlBqS3BQc1FhQWM4VUNiam5IRzlTeXRPSFdROU1COEdB
MVVkSXdRWU1CYUEKRkJFQ3M5RGN4ZVZZRFJMbFNuOWFMc3FJMTFheE1Bc0dBMVVkRHdRRUF3SUY0REFUQmdOVkhTVUV
EREFLQmdncgpCZ0VGQlFjREFUQXlCZ05WSFI4RUt6QXBNQ2VnSmFBamhpRm9kSFJ3T2k4dmJHOWpZV3hvYjNOMEwybHVkR1
Z5CmJXVmthV0YwWlM1amNtd3dJQVlEVlIwUkJCa3dGNElWYTJGc2JXRnVhVzUwWlhKdFpXUnBZWFJsTG1oMU1BMEcKQ1NxR
1NJYjNEUUVCQ3dVQUE0SUVBUUFwaVFERmlUZHB5b2QxRDlYNXlHOFFrMWZBSEJvZnBpS2RhdXd6d0ltdQpjQ0hzMkpDWFlrc
HAvKzYzYWdwazBvSlUvOXcvMk5Nb2hlYjBibnVxcjV6QTBtVEU3d3haNmthTzIzbDBQcStMCnFJS0RETHZQVGl6K0xkOE5FMWp1
WFFrQUxFQnBmdkxNUE15bmV2S3RwaFBQT0x0WGZuNXFLb1lmakViVk90K2kKNFFPTUpaVmNTNkI2eVI2Q1dzWGE2RWdWd
mkxRS9vc0VVRHBRUFluZjlKdE5CMnE5Nll1ZFNYYzM4Y292ckIvVgpxVlU1cWFpYk50Q1IveUZWbFFYVzc5YUNwVTRtTTd4VG5KR
0l1UGpJUEhlZUMxMUNsWExJVTJKR21RVE8waHZlCm1OQmtuWHRpT0NLeTFBbnZnQXA2Q05BNXlKK2p5N2d2OEpXN1Z4N1F

2VFB1SG1IV0NQTVBJbSs0Q2hKd1JaN20KVFNoM2QyOUl4VDJOMjFpRFJqNHNhZUtMSGdRdjVXT2p3UTZEbndBLzRhbTI1VVEv
U0tDa2p2MHEyZkhCY2lJYgpEMTZydXFHcEZValVlNEM2MmRCUzh3dlViS3RTdVJxaVdaUTgzeXJONlFBVTdMSDZ4VE1HN2diTnZ
OTmlzSkR4ClRwbDlGSzBCN1pLaHJsRXcxR1dtRk9LaG5yOXRFS0kyREo4Z2V1eWc1WFNwMmlLclNrOWV5OVBFaDJuZDFVVU0K
dTU3bW1CdXdBVjJXbXU3U0lQNTVQalZEYzY3WDlRNk5Qb09tZmdoRDJ4UzRhZ3pIQ2ZBbTVSZFo4UENuYjlFNgp5SUNkSGNHYjg
1N2FDU3dWK0x4VEdJLzcyQkRtL0ZHRFljU2xnQnpOdGc4aUVnWWVXYXYvV3RyWURsblRkV25LCm1oVlUwMTZ5dGJ0NjhNNkh
udUFFS2RPN3M3L09KWGpCb09SK0ppQ2Z6YUtwb1FlaERldW5KdkNaNEhvQy9BY1cKS0s0VldmTHFneURnMGxZd0FYZXNTQU9
nNnl1aVdYbmVmdHZDTUpnbmVrVVJVTUN1cjM0c1dmWjc1RGJteSs0MQo1T2lyVzYrb1pBenlpZERzSllMZi85R0dvcFIyWDFYelh
wYVZFYXdYaDJ5WlNvV3BuUmR5bnV0L2FxTVJxQnRICnp1Q3YxNEJJSXcvc3QyWSt6MUY0K05iam5RRnZhaGN0Y1U1WFYrTG9R
c0NGbytvRVNRUDcwMFlKKzgzQ1ZyNzUKb0xkSnBVZiszaVpoNXdFSlZJb0tLYVpUS2VGRGxuNFJMbnM1SWVHYVdRVmxzUXZuM
GQxcXpIUVAycThqbnMzbwpPdmZqcGpRaFNQRVRNbUxmOWJYcURSRWZYUHFWS01XRC9ZR2lpUHpiREJJalM3VUtEaHp5clBsR
TVtYjNZVzNjCm54ZnByS3c4Y3BPRDFkbkxGZUlLYUsyeGVSeFQ4aGlwelFWOVlyZWJUK2NmNnIyWTZJRzFYNXE5TDROUmxqUEg
KcGRvVzFhcjJDVTZlK3kwQkVqblR5WE9ueXNSTXVXS1pDTTE0WlBSKzV3RUdZdnEvL0RKOFk0Q2haNXVDc1JQNApnL2k3RVdEe
mo3REhCVzJzSnhVYmZOWms1R2xGVXUxNUpCOE5vRDU4elUyeFVPenpIOHFnQUNiMzNtaVhUNGczCnllRGdWRFFjTXBWaHIzT
mcvU3U4bEJjVWhHQzlSVkcwR2xDZEl3Z3dadzI5ak5JVklDRXR4NVRtOWlWSTRkMlgKQ3I2aTdpQjRZSjRDRnpzeEJSOEdUdXp
GekRCNTN3NVNsa1A3c3praWIwMVQKLS0tLS1FTkQgQ0VSVElGSUNBVEUtLS0tLQo=

x-jws-signature :

eyJhbGciOiJSUzI1NiJ9..c0jgbAraamEer3Iv6qkgjT8Yrov3snbc_uSOTl7gMl_sbRH_fgiL-
4daGKf8j_LKW70xprdgg89I4MEwtAZRfBMwWC2TDlabOwBPJM9rCSQTjiy0E7TulscUxjrcnKzqzD-
y6a77i_EfytqYyhHIT2b5x7xfG0p1MS6vg8HO6cLq5GUaYcvbBVZGnYh_ov0zXDQVJXOFqUMxhQGuX3gckh9elYdQtDM_fMRhaXi
zmR1z9uCrReO5LJJ0x8JZ5cJ9YLJBKs0Nk2R0-
0tZjmm2JpuS3PJpGz0ZW_Qc1zYMBL7tyPyeiSbwj5zr8rXC1Stg0hYjwCcoXYt1g00JMBMw

It is very important that for all REST queries to the NTDSC ATTRAKCIÓ system, the x-jws-signature and x-certificate sections in the HTTP header are filled in correctly!

The digital signature will need to be the key generated by JOSE JWS. The signatory certificate consists of 3 parts: (values from the example above are given in brackets)

header (eyJhbGciOiJSUzI1NiJ9), payload (üres) és signarure(c0jgbAraamEer3Iv6qkgjT8Yrov3snbc_uSOTl7gMl_sbRH_fgiL-
4daGKf8j_LKW70xprdgg89I4MEwtAZRfBMwWC2TDlabOwBPJM9rCSQTjiy0E7TulscUxjrcnKzqzD-
y6a77i_EfytqYyhHIT2b5x7xfG0p1MS6vg8HO6cLq5GUaYcvbBVZGnYh_ov0zXDQVJXOFqUMxhQGuX3gckh9elYdQtDM_fMRhaXi
zmR1z9uCrReO5LJJ0x8JZ5cJ9YLJBKs0Nk2R0-
0tZjmm2JpuS3PJpGz0ZW_Qc1zYMBL7tyPyeiSbwj5zr8rXC1Stg0hYjwCcoXYt1g00JMBMw) .

These three parts, separated by a dot, form the digital signature (JWS token). The value of the x-jws-signature sent to the NTDSC ATTRAKCIÓ system must contain only the header and the signature part. In the header of the digital signature (JWS header), give the type of the encryption algorithm: **RSA256**. The payload part will be generated by the NTDSC ATTRAKCIÓ system from the JSON structure sent in the REST body over HTTP (this is why the payload part may be left empty in the signature sent), and then checked whether the data sent and the signature sent in the request are valid together or not. If the signature is not satisfactory, the system will indicate it in the list of errors to the sender.

If the signature check is correct, the NTDSC ATTRAKCIÓ system will start the semantic check of the submitted data.- If the system finds an error, it puts all the errors on an list of errors. When the sending system queries the status of the submitted data with the processing identifier, it will see a list of successfully and unsuccessfully processed data. The list of errors for the data that have been processed unsuccessfully will contain the data to be corrected. Failed messages are corrected and resent to the TSS system for the NTDSC ATTRAKCIÓ system.

If the request contains only correctly submitted data, i.e. data that can be processed, the message will have the status TELJESEN_SIKERES. In this case, the list of failed messages will be completely empty.

If the request contains only incorrectly submitted/unprocessable data, the message status will be set to TELJESEN_HIBAS. In this case the list of successful messages will be completely empty. In this case, all unsuccessful requests must be resubmitted with corrections.

If the request contains both correctly and incorrectly submitted data, the status of the message will be RESZBEN_SIKERES. In such cases, neither the list of unsuccessful messages, nor the list of successful messages will be empty. In this case, all unsuccessful requests must be resubmitted with corrections.

If the status check returns with UJRA_KULDENDO, it means that the message could not be processed despite being received due to some technical error or data loss, so the sending system must resend the message without any changes.

5.2. Error keys on the TSS interface

Error key	Error message (Hungarian)	Error message (English)
NotEmpty	Nem lehet üres értékű a mező!	The field cannot be empty!
NotNull	Nem lehet null értékű a mező!	The field cannot be null!
Size	Mező hossza nem megfelelő!	Field length is incorrect!
Max	Az érték túllépte a maximumot!	Value exceeded maximum!
Negative	Nem lehet negatív a mező értéke!	The value of this field cannot be negative!
Zero	A mező értéke nem lehet nulla!	Field value cannot be zero!
MissmatchType	A mező típusa nem megfelelő!	Field type is incorrect!
ErrorEnum	A mező hibás értéket tartalmaz! A mező értékkészlete: <enum>	This field contains an invalid value! The value set for this field is <enum>
DuplicatedId	A beküldött üzenetben nem lehet egynél többször ugyanaz a rendelés azonosító!	The specified value cannot be present multiple times in a message!
UniqueConstraint	A megadott érték már szerepel az adatbázisban!	The specified value is already in the database!
Future	A mező időpontja nem lehet jövőbeli!	Field datetime value cannot be in the future!
Past	A mező időpontja nem lehet múltbeli!	Field datetime value cannot be in the past!
Conflict	A mező értéke ellentmondásban áll a függőségeivel!	The value of the field conflicts with its dependencies!
ReferenceConflict	A hivatkozott azonosító nem található!	The referenced ID could not be found!
MissingProgram	A hivatkozott program nem található!	The referenced program could not be found!
NotFoundInDb	Nem található az adatbázisban!	Not found in the database!
SelfReference	A hivatkozott program saját magára mutat!	The referenced program referencing itself!

Error key	Error message (Hungarian)	Error message (English)
CombinedReference	A kombinált program egy másik kombinált programot tartalmaz!	The combined program contains another combined program!
MissingIdSzolgHelyAzonositoSzamAttrakcioban	Nem található a Szolgáltatási hely azonosító szám az ATTRAKCIÓ ágazatban!	The identification number of the 'Szolgáltatási hely' is not found in 'ATTRAKCIÓ'!
MissmatchSzolgaltatoAdatokAdoszam	Nem található Szolgáltató adószám az ATTRAKCIÓ ágazatban!	The tax number of the 'Szolgáltató' is not found in 'ATTRAKCIÓ'!
MissingTSSSystemInAttrakcio	Nem található ilyen TSS rendszer azonosító a ATTRAKCIÓ-ban!	No such TSS system ID found in 'ATTRAKCIÓ'!
ErrorKuldorendszer	Hiba a küldő rendszer adatokban!	Error in sending system data!
ErrorSzolgaltato	Hiba a szolgáltató adatokban!	'Szolgáltató' data error!
NotFoundInDbAttrakcio	Attrakció nem található az adatbázisban!	Attraction not found in the database!
NotFoundInDbSzHely	Szolgáltatási hely nem található az adatbázisban!	Szolgáltatási hely' not found in the database!
NullJwsToken	A kérésben üres a digitális aláírás!	The digital signature in the request is empty!
JwsTokenError	Digitális aláírás hiba!	Digital signature error!
InvalidAudienveJwsToken	Nem engedélyezett digitális aláírás hallgatóság!	Digital signature audience not allowed!
InvalidJwsToken	Nem érvényes digitális aláírás!	Digital signature is not valid!
ExpiredJwsToken	Lejárt digitális aláírás!	Digital signature expired!
NullCertificate	A kérésben üres a tanúsítvány!	The certificate is empty in the request!
ErrorReadCertificate	Tanúsítvány beolvasás hiba!	Certificate retrieval error!
ErrorCertificate	Tanúsítvány általános hiba!	Certificate general error!
ErrorCheckCertificateType	Tanúsítvány típusa nem meghatározható!	Certificate type cannot be determined!
ErrorCertificateChain	Tanúsítványlánc ellenőrzés hiba!	Certificate chain check error!
ErrorCertificateChainCheck	Tanúsítványlánc ellenőrzés sikertelen!	Certificate chain check failed!
MissmatchCertificateCnCheck	A tanúsítvány CN és a body-ban küldött érték nem egyezik!	The certificate CN and the value sent in the body do not match!
CheckBeforeDateCertificateError	Tanúsítvány még nem érvényes!	Certificate not yet valid!
ExpiredCertificate	Lejárt tanúsítvány!	Expired certificate!
NotTrustedCertificate	Nem megbízható tanúsítvány!	Untrusted certificate!
JsonSyntaxError	JSON szintaktikai hiba: <hiba>	JSON syntax error: <error>

6. Appendices

6.1. Requesting a certificate

6.1.1. Standards and Recommendations

The following table summarises the list of standards and recommendations referenced in this document.

Keyword	
RFC 5280 **X.509** **CRL**	Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile https://www.ietf.org/rfc/rfc5280.txt
RFC 6818 **X.509** **CRL**	Updates to the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile https://tools.ietf.org/html/rfc6818
RFC 6960 **OCSP**	X.509 Internet Public Key Infrastructure Online Certificate Status Protocol (OCSP) https://tools.ietf.org/html/rfc6960
RFC 2986 **PKCS#10**	PKCS #10: Certification Request Syntax Specification Version 1.7 https://tools.ietf.org/html/rfc2986

6.1.2. Creating a certificate signing request in a Windows environment

For testing purposes, in a Windows operating system environment, you can use the command line using **certreq** utility to create a certificate signing request

https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/certreq_1

Creating a test certificate signing request

For the certificate signing request example, a configuration file similar to the one below should be created:

```
;
; SampleAuthentication.ini
; Content of the certificate signing request file used for testing
;
[NewRequest]
Subject = "CN=12345678"
FriendlyName = "Példa Tanúsítvány"
KeySpec = 1
KeyLength = 4096
KeyUsage = "CERT_DIGITAL_SIGNATURE_KEY_USAGE | CERT_NON_REPUDIATION_KEY_USAGE | CERT_KEY_ENCIPHERMENT_KEY_USAGE"
HashAlgorithm = SHA256
KeyAlgorithm = RSA
Exportable = TRUE
MachineKeySet = TRUE
SMIME = FALSE
PrivateKeyArchive = FALSE
UserProtected = FALSE
UseExistingKeySet = FALSE
ProviderName = "Microsoft Enhanced RSA and AES Cryptographic Provider"
```

```
ProviderType = 24
RequestType = PKCS10

[EnhancedKeyUsageExtension]
OID=1.3.6.1.5.5.7.3.2
```

It is important to note that in the above example configuration file, the Exportable=TRUE parameter allows the created key pair to be moved to another computer, and the MachineKeySet=TRUE parameter allows the key pair to be created in the computer's certificate repository, where the issued certificate must be loaded.

The mandatory data in the issued certificate signing request is the registration number of the service location for which the certificate signing request is issued. This data must be entered in the Subject CN field as shown above, the identification number of the service location in the example, therefore, is “12345678”.

To generate a certificate signing request from the command line, use the following command:

```
CertReq –New SampleAuthentication.ini SampleAuthentication.csr
```

6.1.3. Creating a certificate signing request in a Linux environment

For testing purposes, in a Linux operating system environment, you can use the command line using **openssl** utility to create a certificate signing request

https://www.openssl.org/docs/manmaster/man1/openssl.html

Creating a test certificate signing request

For the certificate signing request example, a configuration file similar to the one below should be created:

```
;
; SampleAuthentication.conf
; Content of the certificate signing request file used for testing
;
[ req ]
default_bits      = 4096
default_md        = sha256
prompt            = no
encrypt_key       = no
distinguished_name = req_distinguished_name
req_extensions    = req_extensions
```

```
[ req_distinguished_name ]
CN              = 12345678

[ req_extensions ]
keyUsage        = digitalSignature, keyEncipherment, nonRepudiation
extendedKeyUsage  = clientAuth
```

When using a command line, the certificate signing request needs to be created with the following command:

```
openssl req -newkey rsa:4096 -keyout SampleAuthentication.key -config SampleAuthentication.conf -out SampleAuthentication.csr
```

The mandatory data in the issued certificate signing request is the identification number of the service location for which the certificate signing request is issued. This data must be entered in the Subject CN field as shown above, the identification number of the service location in the example, therefore, is “12345678”.

To check the correctness of the data in the certificate signing request file, use the following command:

```
openssl req -in SampleAuthentication.csr -noout -text
```

6.2. Interface message examples

The following sample messages were submitted in a test system of the interface and therefore the identifiers in them do not correspond to any existing system, service provider or organization. The code sections are formed according to the syntax of cURL command-line tool.

6.2.1. Sales for an independent programme

```
curl --location --request POST 'https://tss.ntaktst.hu/api/jegyertekesites' --header 'Content-Type: application/json' --header 'x-jws-signature: eyJhbGciOiJSUzI1NiJ9..HG93syARC9SUnL9XT9ECZkozuo5O7kOzRgehcwRQH2L8kZg82kSy2tIPjYC1TBfR7OvSVqK4-dudehCTZniPABdwlKi7XmzkNHe4kbhSfvRNhDiEtsioka7Sk3ObAYarenLE7XMPY8mu5Mt88pLgk8NmTudC6Th88NJ3fJn8sRcC6LWRG9cA5KXS1G0OY5IX4DuiX9VIfSzsvlFQweN7wkv6zIVy0V8gaWx9UwfdIq_IYa27BmY3_8rGRyMxKZ0cOX9vZ0ltLrb_lxWPSgXoI1yuYDfPAmE3mlZ1q30lYDEHvPlSD1o0o1brhPrg3S6zxEb-R4-R70a5FBXARwSlv1uWEz0k1uG6IC_HSbAsi_j3fnj9YRf7kI2EE7jWFXIQmyaCpQ8tfnYe4zHFt-QAMZ-f9Ai46XuZDUM8gjyp9nEIIbxOZm6FDHIX9q_PYVp6nNHyi9E0-cyUWkItyaYcqabnobJ2cTh4EFBe_5xlO6DLx1glUhVhkuILGZyyxI7umUaUkFjkjdBy1LxROFPK5xSRLUeEzNVNVaZ6EVANEmOGM0_2YJJCJEa8rhKxR-XOUZ77OVakKrjUH7ufEemoicrrEIUJ9UI1hu0Hru33NPi_a_ENMXsUJ7gGzK0I1pcVZ5gObObDqcXtD5XP_NQMzNYf5SJHNioxB_GiOxluIyQ' --header 'x-certificate:LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tDQpNSUlIZlRDQ0JXV2dBd0lCQWdJUkFVZ2MrSnRGa2pKR3NnYjlqamlHaHpnd0RRWUpLb1pJaHZjTkFRRUxCUUF3DQpnWTB4TWpBd0JnTlZCQU1NS1ZSRlUxcFVJRTVVUVVzdFEwRWdRMnhwWlc1MElFRjFkR2dnVTJsbmJtbHVaeUJEDQpRU0F5TURJeU1Sc3dHUVlEVlFRTERCSlVSVk5hVkNCT1ZFRkxMVU5CSURJd01qSXhHakFZQmdOVkJBb01FVlJGDQpVMXBVSUVSQlZGSkJTeUJMWm5RdU1SRXdEd1lEVlFSERBaENkV1JoY0dWemRERUxNQWtHQTFVRUJoTUNTRlV3DQpIaGNOTWpJeE1qQXlNVFl3TmpJd1doY05NalF4TWpBeE1UWXdOakl3V2pDQm5URUxNQWtHQTFVRUJoTUNTRlV4DQpHREFXQmdOVkJBY01EMEoxWkdGd1pYTjBJQzBnU1VsSkxqRU5NQXNHQTFVRUVRd0VNVEF6TkRFUk1BOEdBMVVFDQpEQWd5TWpBd05ESXpNekVSTUE4R0ExVUVDd3dJVXpRNFNrRkdPRWN4RURBT0JnTlZCQVVUQjBwbFoza3VhSFV3DQpnZ0lpTUEwR0NTcUdTSWIzRFFFQkFRVUFBNElDRHdBd2dnSUtBb0lDQVFDcEVMTGp0eDBERGtOdmUzd1k5TTJvDQo0SjdMT0c1bkVyY1dwSkF4ZjZ4aThxVHlTRW11NTBYQmlUa0FLNUNWZVRBOThvNEEwWUs5OCtuMzB3ZHF
```

```
    "szolgHelyRegisztraciosSzam": "MK22004233"
  },
  "kuldoRendszerAdatok": {
    "tssRendszerNTAKAzonosito": "Jegy.hu",
    "tssRendszerVerzioszam": "1"
  },
  "uzenetAdatok": {
    "uzenetKuldesIdeje": "2022-12-02T17:39:12.289+01:00"
  },
  "programAdatok": [
    {
      "helyszinAdatok": {
        "helyszin": "test_helyszin",
        "iranyitoszam": "1111"
      },
      "programAlkategoria": "TEMATIKUS_BEMUTATOHELY_LATOGATAS",
      "programFokategoria": "VEDETT_TERMESZETI_TERULET",
      "programGyakorisaga": "ALLANDO",
      "onlineProgram": false,
      "programKezdete": "2022-04-01T00:00:00+02:00",
      "programNeve": "Minimális program",
      "programTipusa": "ONALLO_PROGRAM",
      "programVege": "2022-04-30T23:59:59.99+02:00",
      "tssProgramAzonosito": "P001",
      "utolsoModositasIdeje": "2022-04-06T09:04:44.65+02:00"
    }
  ],
  "tranzakcioAdatok": [
    {
      "tranzakcioAzonosito": "T-1a605802-50c2-4347-88c2-6304cf96cab5",
      "tranzakcioIdopontja": "2022-04-06T09:04:44.65+02:00",
      "osszesitett": false,
      "tranzakcioVegosszege": 501,
      "ertekesitesiCsatorna": "AUTOMATA",
      "fizetesiModok": [
        {
          "fizetesiMod": "KESZPENZHUF",
          "fizetettOsszegHUF": 500
        },
        {
          "fizetesiMod": "KEREKITES",
          "fizetettOsszegHUF": 1
        }
      ],
      "latogatoiAdatok": {
        "kulfoldi": false,
        "latogatokLakohelye": "1111"
      },
      "eladottJegyek": [
        {
          "afaKategoria": "C_27",
          "azonnalFelhasznalt": false,
          "bruttoAr": 1001,
          "ervenyessegKezdete": "2022-04-06T09:04:44.62+02:00",
          "jegyAzonosito": "J-1a605802-50c2-4347-88c2-6304cf96cab5",
          "jegyErvenyessegTipusa": "NAPI_JEGY",
          "jegyMegnevezes": "J001",
          "kedvezmenyek": [],
          "korcsoport": "FELNOTT",
          "maxBelepesekSzama": 1,
          "ntakRendszerKategoria": "EGYENI",
          "programazonosito": {
```

```
            "tssProgramAzonosito": "P001",
            "utolsoModositasIdeje": "2022-04-06T09:04:44.65+02:00"
          },
          "szemelyekSzama": 1
        }
      ],
        "egyebTetelek": [
        {
          "tetelAzonosito": "E-1a605802-50c2-4347-88c2-6304cf96cab5",
          "tetelMegnevezes": "Hétvégi kedvezmény",
          "kategoria": "KEDVEZMENY",
          "tetelszam": 1,
          "bruttoEgysegAr": -500,
          "afaKategoria": "C_27"
        }
      ]
    }
  ]
    }'
```


6.2.2. Sales for a combined programme


```
curl --location --request POST 'https://tss.ntaktst.hu/api/jegyertekesites' --header 'Content-Type: application/json' --header 'x-
jws-signature: eyJhbGciOiJSUzI1NiJ9..Q2QKfLxrfwWVqmtmVjorRAAJFhNloOAWUeHdFoa3q1vS5ogl-
zNnqwg6axz5GuJGBxakCy8c_JxjUjmT8wGQGLLMSWI5vGnb1cqHynUzrTSF8p0o90jcmqFnWQizKEs94fUOpc7dgzE_LEJwlNZKCT8E2
_ttOITsTpOLcB8qm13Ur4Iy_OYxX1YZCiZ_5vtoq82U4CfDVk7lN8oWga5vBFgIGuL0BHRksKiH7zLcIwZ06ooVtQgnkD6wl9eOCBOWki5
zybnv-
Px4wfGCW0KemZzcABPDzAeG9PE9X781puefaWVtFhJ3YpxIREowlyu58fZ5j_YRSoCPOIUfQVULVz37LFFPl8O8SXTlVM_XZz49Szbfat
m8qx0y6wSTM8lMLJPJADxv2bSk-pk9fE6ofgGjXCsWBlRP29GkBLao-
CHNyRavwSapJcTABleI7Agr31JI8X6Z38FyiUnVHU04y8j_Lgyx8FmfjI57ETnNA8jG0Gr3n7at74anp_58SleO7EdYrWKpfV2k3u7j7WMY
VUOL7cTb43VtqiSp3AMd3TeEyp04H7WKfl5pFMfsIJtQth8B3SfMXy2Tb25FAd690ksvV2T1Cqk4LLQ3ByPCUrUv7JICxv4mI8fUzmxcAq
Lf7juK_UCXE4zEH_dxvaO89O6B_WoVbFCvL-522xV_w_Y' --header 'x-certificate:
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tDQpNSUlIZlRDQ0JXV2dBd0lCQWdJUkFVZ2MrSnRGa2pKR3NnYjlqamlHaHpnd0RRWUpL
b1pJaHZjTkFRRUxCUUF3DQpnWTB4TWpBd0JnTlZCQU1NS1ZSRlUxcFVJRTVVUVVzdFEwRWdRMnhwWlc1MElFRjFkR2dnVTJsbmJtb
HVaeUJEDQpRU0F5TURJeU1Sc3dHUVlEVlFRTERCSlVSVk5hVkNCT1ZFRkxMVU5CSURJd01qSXhHakFZQmdOVkJBb01FVlJGDQpVMX
BVSUVSQlZGSkJTeUJMWm5RdU1SRXdEd1lEVlFRSERBaENkV1JoY0dWemRERUxNQWtHQTFVRUJoTUNTRlV3DQpIaGNOTWpJeE1q
QXlNVFl3TmpJd1doY05NalF4TWpBeE1UWXdOakl3V2pDQm5URUxNQWtHQTFVRUJoTUNTRlV4DQpHREFXQmdOVkJBY01EMEoxW
kdGd1pYTjBJQzBnU1VsSkxqRU5NQXNHQTFVRUVRd0VNVEF6TkRFUk1BOEdBMVVFDQpZUXdJTVRJek5EVTJNekl4R2pBWUJnTlZCQ
W9NRVZSVFV5QlNReUJOU3pJeU1EQTBNak16TVJFd0R3WURWUVFEDQpEQWd5TWpBd05ESXpNekVSTUE4R0ExVUVDd3dJVXpRN
FNrRkdPRWN4RURBT0JnTlZCQVVUQjBwbFoza3VhSFV3DQpnZ0lpTUEwR0NTcUdTSWIzRFFFQkFRVUFBNElDRHdBd2dnSUtBb0lDQV
FDcEVMTGp0eDBERGtOdmUzd1k5TTJvDQo0SjdMT0c1bkVyY1dwSkF4ZjZ4aThxVHlTRW11NTBYQmlUa0FLNUNWZVRBOThvNEEw
WUs5OCtuMzB3ZHFFVkM3DQpvcTVkZ3VVakt6M1RwRlpKTWR3Qzc1NGNSVXYyTjJwRk0ySzJ2bFdtd0pjcVBLMlZVanNMR0ZJd3EvN
HYwMWZTDQpuV3pnNGUxZHJjOVZZZVN4UnA0STZBUm1PQnlaQjNwbDdXYmZiQ1YrZmU1TFFYTjJ3QVhCV2ZKb2xrS3lpaFRnDQpx
OEoxODkvOXBHckRCbWs0dFVUTkNiaXVpZklOM3NWeGcrUHZUTzZVN1ExTVM2MFd3VW5JL1NWajJnUm9TTTY1DQpsUjRYU25vSH
BLcDB5SCtuWGIvdnAwZTc5T04zWnVYWnB3aFlidGhXWllNcFlXN3JNcUVZZW43S0MzZjh1QmlEDQo5QlVZZWoxa3M0YWc1V0xXS0J
hWW9PQkRucHJDWlZaSFRHa0xrTnQ3UHRwVVlxOWdIVW8rRU95YkZmNENvSk1JDQpOZ1VWNERxd2ZUNDQrWkROSmF5M3F6Tlp
iNEQvemtJb1N3bzRFRUNyS1N2Vk1UVGo0Ry9lb2NmQk1DRWI0S1BSDQpONkpQT2tldzRwbGdYaDR5QllkSy82amdHZEdRWFNORE
VGVWorZjdTMXJ0SURrcnpjeFhmL3pYaEdWMzl5cDVSDQpDblNTZTROaThvYzNjUmR5dVZGOHZaalVCSzNURERCbXlZdzZvNFUyWGR
Ka2F5ZXRVRCtwamhnRUUrc1ViNHM5DQpzL3doWnJheW5WaW5KYWp2WGRzVjZCblVhMGMwL3ZlemZFenQ2Y2dERTUrSGh1WlB
xaVRpY2dXVUM1NlBQODR5DQpaYUVJQzNPSUN1RTNoSWRkVTh4SWZRSURBUUFCbzRJQnhEQ0NBY0F3Q1FZRFZSMFRCQUl3QUR
CckJnTlZIUjhFDQpaREJpTUdDZ1hxQmNobHBvZEhSd09pOHZkR1Z6ZW5RdWJuUmhhMk5oTG1oMUwyNTBZV3RqWVdOc2FXVnVkR
0YxDQpkR2h6YVdkdWFXNW5ZMkV5TURJeUwyTnliQzI1ZEdGclkyRmpiR2xsYm5SaGRYUm9jMmxuYm1sdVoyTmhNakF5DQpNaTVq
Y213d2djUUdDQ3NHQVFVRkJ3RUJCSUczTUlHME1Hc0dDQ3NHQVFVRkJ6QUNobDlvZEhSd09pOHZkR1Z6DQplblF1Ym5SaGEyTmh
MbWgxTDI1MFlXdGpZV05zYVdWdWRHRjFkR2h6YVdkdWFXNW5ZMkV5TURJeUwyTmhhWE56DQpkV1Z5TDI1MFlXdGpZV05zYVd
WdWRHRjFkR2h6YVdkdWFXNW5ZMkV5TURJeUxtTmxjakJGQmdnckJnRUZCUWN3DQpBWVk1YUhSMGNEb3ZMM1JsYzNwMExtNT
BZV3RqWVM1b2RTOXVkR0ZyWTJGamJHbGxiblJoZFhSb2MybG5ibWx1DQpaMk5oTWpBeU1pOXZZM053TUE0R0ExVWREd0VCL3d
RRUF3SUY0REFhQmdOVkhTQUVFekFSTUE4R0RTc0dBUVFCDQpnOE1hQVFNQkF3SXdFd1lEVlIwbEJBd3dDZ1lJS3dZQkJRVUhBd0l3S
HdZRFZSMGpCQmd3Rm9BVVRPVXFmVWQ5DQp1M1N1Z09rZHV6eCtQVDBJVExFd0hRWURWUjBPQkJZRUZQNnZYRFc0ZXJBVGhkT
mlNaEdRVHlxNWljcEtNQTBHDQpDU3FHU0liM0RRRUJDd1VBQTRJQ0FRQllkRzU2c20wR0FncU8xT3hHRFhYT3ZiWnYwbkdMcWFhR
VY4T0h3L3pPDQo3SVlLZWVtWmRicURONG1rdFRsSzkyLytaN2dsTGV3MzBrbExSVklDRGRxWmpRR0ZXVDhnMkR3d2pOV3cydWxvD
Qp1Njl0V1Zjd1pJcEQwU2RwMTdBWVB5WFliZ3R5VG41aldONldnSk83VTRxVHA2MkNOSUJJa1JnQzNJR2ROK1pmDQpVQm9lNVRiN
```

```
HkyMEZKUWVxQXYwbWhuajdxOXRMZEdCSHYzcTQ0MmhmeFNGRTJKVHNHNTRUZlFGeHVjeDlySTlJDQpVdXZRWHo0MW1hcjdOV
29QN0YyZ2xqcllIOWt1OEVpTWNYWjZEZ2pEZ1lLL2F5QitPQTdJL3dad3ZubHFaRkdNDQpGL2c0aHU3OEpsd0N2aG14cDJ5Yld3c1JSR
Ud0M0RtSDBpbTM5K2dPdzZUTDhnL1FlbXNKeFN6dU9FRlo1S3ZlDQpTcWZjN2FBT1o2ZEZJTnhmQWV1OGJBSTNzTXZiQWp3aW1Z
UW41WGRLSjJUSnEwOGdGWkkrODk1Y3ZadGJtZmUxDQpZdjd4bXFFaGlOamxLNGd3azRGVW9mYVlmejJPcnpPMXhWSzBubnZ1Mk
x2M0FyWFZzcUhMd3Y0dTFlcUVmcDBBDQpDc05lU01QeFY5T2NzS010UHZ1L0VFQWpNVm44Z1FzR0s4N3dHRHdwemZPVkFKUUR
pWkRuSzdISVUxNGxnZVhnDQozQlN5TFM2d1lyR0Q3U3FFK0xrMVU4cHh0UU1WallOVE9lNzBFL0FxV1EreCtZbHQ1SEN6azkxMWQ4
OVF1V3VuDQpURzFIT1c1TnlJUU9TZnJDV1k2amFaM2NoZWxvQUFtVzNrVmt3YTV1S2x3Vk5kYTNweVhZM2RHRGRwcE9RWkE1DQp
5Zz09DQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tDQo=' --data-raw '{
  "szolgaltatoAdatok": {
    "adoszam": "12345632243",
    "szolgHelyRegisztraciosSzam": "MK22004233"
  },
  "kuldoRendszerAdatok": {
    "tssRendszerNTAKAzonosito": "Jegy.hu",
    "tssRendszerVerzioszam": "1"
  },
  "uzenetAdatok": {
    "uzenetKuldesIdeje": "2022-12-02T17:42:37.979+01:00"
  },
  "programAdatok": [
    {
      "helyszinAdatok": {
        "helyszin": "test_helyszin",
        "iranyitoszam": "1111"
      },
      "programAlkategoria": "TEMATIKUS_BEMUTATOHELY_LATOGATAS",
      "programFokategoria": "VEDETT_TERMESZETI_TERULET",
      "programGyakorisaga": "ALLANDO",
      "onlineProgram": false,
      "programKezdete": "2022-04-01T00:00:00.0Z",
      "programNeve": "Első alprogram",
      "programTipusa": "ONALLO_PROGRAM",
      "programVege": "2022-04-30T23:59:59.999Z",
      "szolgHelyProgramAzonosito": "",
      "tssProgramAzonosito": "P002",
      "utolsoModositasIdeje": "2022-04-06T09:04:44.652Z"
    },
    {
      "helyszinAdatok": {
        "helyszin": "test_helyszin",
        "iranyitoszam": "1111"
      },
      "programAlkategoria": "TEMATIKUS_BEMUTATOHELY_LATOGATAS",
      "programFokategoria": "VEDETT_TERMESZETI_TERULET",
      "programGyakorisaga": "ALLANDO",
      "onlineProgram": false,
      "programKezdete": "2022-04-01T00:00:00.0Z",
      "programNeve": "Második alprogram",
      "programTipusa": "ONALLO_PROGRAM",
      "programVege": "2022-04-30T23:59:59.999Z",
      "szolgHelyProgramAzonosito": "",
      "tssProgramAzonosito": "P003",
      "utolsoModositasIdeje": "2022-04-06T09:04:44.652Z"
    },
    {
      "helyszinAdatok": {
        "helyszin": "test_helyszin",
        "iranyitoszam": "1111"
      },
      "programAlkategoria": "TEMATIKUS_BEMUTATOHELY_LATOGATAS",
      "programFokategoria": "VEDETT_TERMESZETI_TERULET",
      "programGyakorisaga": "ALLANDO",
```

```
      "onlineProgram": false,
      "programKezdete": "2022-04-01T00:00:00.0Z",
      "programNeve": "Kombinált program",
      "programTipusa": "KOMBINALT_PROGRAM",
      "programVege": "2022-04-30T23:59:59.999Z",
      "szolgHelyProgramAzonosito": "",
      "tssProgramAzonosito": "P004",
      "utolsoModositasIdeje": "2022-04-06T09:04:44.652Z",
      "kapcsolodoProgramok": [
        {
          "tssProgramAzonosito": "P002",
          "utolsoModositasIdeje": "2022-04-06T09:04:44.652Z"
        },
        {
          "tssProgramAzonosito": "P003",
          "utolsoModositasIdeje": "2022-04-06T09:04:44.652Z"
        }
      ]
    }
  ],
  "tranzakcioAdatok": [
    {
      "tranzakcioAzonosito": "T-ca062050-1fbe-434c-bebe-533737a4d32b",
      "tranzakcioIdopontja": "2022-04-06T09:04:44.65+02:00",
      "osszesitett": false,
      "tranzakcioVegosszege": 501,
      "ertekesitesiCsatorna": "AUTOMATA",
      "fizetesiModok": [
        {
          "fizetesiMod": "KESZPENZHUF",
          "fizetettOsszegHUF": 500
        },
        {
          "fizetesiMod": "KEREKITES",
          "fizetettOsszegHUF": 1
        }
      ],
      "latogatoiAdatok": {
        "kulfoldi": false,
        "latogatokLakohelye": "1111"
      },
      "eladottJegyek": [
        {
          "afaKategoria": "C_27",
          "azonnalFelhasznalt": false,
          "bruttoAr": 1001,
          "ervenyessegKezdete": "2022-04-06T09:04:44.62+02:00",
          "jegyAzonosito": "J-ca062050-1fbe-434c-bebe-533737a4d32b",
          "jegyErvenyessegTipusa": "NAPI_JEGY",
          "jegyMegnevezes": "J001",
          "kedvezmenyek": [],
          "korcsoport": "FELNOTT",
          "maxBelepesekSzama": 1,
          "ntakRendszerKategoria": "EGYENI",
          "programazonosito": {
            "tssProgramAzonosito": "P004",
            "utolsoModositasIdeje": "2022-04-06T09:04:44.652Z"
          },
          "szemelyekSzama": 1
        }
      ],
        "egyebTetelek": [
```

```
        {
          "tetelAzonosito": "E-ca062050-1fbe-434c-bebe-533737a4d32b",
          "tetelMegnevezes": "Hétvégi kedvezmény",
          "kategoria": "KEDVEZMENY",
          "tetelszam": 1,
          "bruttoEgysegAr": -500,
          "afaKategoria": "C_27"
        }
      ]
    }
  ]
}'
```

6.2.3. Ticket validation

```
curl --location --request POST 'https://tss.ntaktst.hu/api/jegyervenyesites' --header 'Content-Type: application/json' --header 'x-jws-signature: eyJhbGciOiJSUzI1NiJ9..Zs7wl331gDQWjad_dHPYv54FIltsyfEQO0a6WaJR3CpaabXA3cSqQf8-9zDT4hLNSrslwMCvcx76NQu1TnmgVnCTGhMqbv7qSs9kgVqlcWj9Gfh6cdhLKgWRVlWY2N-4QiRA-07zcwdJPScNV2mwuIvuUNf_ojJAUTsOofo_yQQ-mEXSGssa9GrtXfYM7hPBm3GIgtzgm_UkBNybsEYwT9aDybXpNOjzdjA-Z1ghSvjvrjnMT4Bh2F7xT2uKi-PU_oSxrabkqvyVhdFm04XsOk1rqLdSlXZkr7sjlwWz4Y9nMsDI65ghTQVKvEKTzRyIcp4JbMnhp1Wk04DQYJ5d1DgLAyBPPkH9bhwghBj8DVw6335-er4Fic4nAgCO-uShiPapy9TFEuDpZpRveKR-E0TENnR4a4YG994C7UmkI9SLLRJ_6Ko9_HFfpy-jHXDvCmXsDOi-5gfEzRl6BS0nVLwpS2N6uwBaMAWB9Iy_8HoqvCP3FHkKQxBGT3MS_IQDIoDEPgaUYbOuVMUfgon1A_gMNOQ-1pUv3d6rZt_H6vYhz4ob4IWr7bDfio5TWnAhn7gmeYXF8OuMPTCCWyiLjtIHw4Aio0x9IIn3s4YZnTNRqXqNAnOh82n4zpeYSEmJlmvkJRAaw2JOAPe1_A8hX4OQELB4p-2EF6yeLoXQeM0' --header 'x-certificate: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tDQpNSUlIZlRDQ0JXV2dBd0lCQWdJUkFVZ2MrSnRGa2pKR3NnYjlqamlHaHpnd0RRWUpLb1pJaHZjTkFRRUxCUUF3DQpnWTB4TWpBd0JnTlZCQU1NS1ZSRlUxcFVJRTVVUVVzdFEwRWdRMnhwWlc1MElFRjFkR2dnVTJsbmJtbHVaeUJEDQpRU0F5TURJeU1Sc3dHUVlEVlFRTERCSlVSVk5hVkNCT1ZFRkxMVU5CSURJd01qSXhHakFZQmdOVkJBb01FVlJGDQpVMXBVSUVSQlZGSkJTeUJMWm5RdU1SRXdEd1lEVlFRSERBaENkV1JoY0dWemRERUxNQWtHQTFVRUJoTUNTRlV3DQpIaGNOTWpJeE1qQXlNVFl3TmpJd1doY05alF4TWpBeE1UWXdOakl3V2pDQm5URUxNQWtHQTFVRUJoTUNTRlV4DQpHREFXQmdOVkJBY01EMEoxWkdWemRERUxNQWtHQTFVRUJoTUNTRlV3DQpnZ0lpTUEwR0NTcUdTSWIzRFFFQkFRVUFBNElDRHdBd2dnSUtBb0lDQVFDcEVMTGp0eDBERGtOdmUzd1k5TTJvDQo0SjdMT0c1bkVyY1dwSkF4ZjZ4aThxVHlTRW11NTBYQmlUa0FLNUNWZVRBOThvNEEwWUs5OCtuMzB3ZHFFVkM3DQpvcTVkZ3VVakt6M1RwRlpKTWR3Qzc1NGNSVXYyTjJwRk0ySzJ2bFdtd0pjcVBLMlZVanNMR0ZJd3EvNHYwMWZTDQpuV3pnNGUxZHJjOVZZZVN4UnA0STZBUm1PQnlaQjNwbDdXYmZiQ1YrZmU1TFFYTjJ3QVhCV2ZKb2xrS3lpaFRnDQpxOEoxODkvOXBHckRCbWs0dFVUTkNiaXVpZklOM3NWeGcrUHZUTzZVN1ExTVM2MFd3VW5JL1NWajJnUm9TTTY1DQpsUjRYU25vSHBLcDB5SCtuWGlvdnAwZTc5T04zWnVYWnB3aFlidGhXWllNcFlXN3JNcUVZZW43S0MzZjh1QmlEDQo5QlVZZWoxa3M0YWc1V0xXS0JhWW9PQkRucHJDWlZaSFRHa0xrTnQ3UHRwVVlxOWdIVW8rRU95YkZmNENvSk1JDQpOZ1VWZVoxS0JiBWgxTDI1MFlXdGpZV05zYVdWdWRHRjFkR2h6YVdkdWFXNW5ZMkV5TURJeUwyTmhhWE56DQpkV1Z5TDI1MFlXdGpZV05zYVdWdWRHRjFkR2h6YVdkdWFXNW5ZMkV5TURJeUxtTmxjakJGQmdnckJnRUZCUWN3DQpBWVk1YUhSMGNEb3ZMM1JsYzNwMExtNTBZV3RqWVVdOc2FXVnVkR0YxDQpkR2h6YVdkdWFXNW5ZMkV5TURJeUxtTmxjakJGQmdnckJnRUZCUWN3DQpBWVk1YUhSMGNEb3ZMM1JsYzNwMExtNTBZV3RqWVVkR0ZyWTJGamJHbGxiblJoZFhSb2MybG5ibWx1DQpaMk5oTWpBeU1pOXZZM053TUE0R0RxVWREd0VCL3dRRUF3SUY0REFhQmdOVkhTQUVFekFSTUE4R0RTc0dBUVFCDQpnOE1hQVFNQkF3SXdFd1lEVlIwbEJBd3dDZ1lJS3dZQkJRVUhBd0l3SHdZRFZSMGpCCQmd3Rm9BVVRPVXFmVWQ5DQp1M1N1Z09rZHV6eCtQVDBJVExFd0hRWURWUjBPQkJZRUZQNnZYRFc0ZXJBVGhkTmlNaEdRVHlxNWljcEtNQTBHDQpDU3FHU0liM0RRRUJDd1VBQTRJQ0FRQllkRzU2c20wR0FncU8xT3hHRFhYT3ZiWnYwbkdMcWFhRVY4T0h3L3pPDQo3SVlLZWVtWmRicURONG1rdFRsSzkyLytaN2dsTGV3MzBrbExSVklDRGRxWmpRR0ZXVDhnMkR3d2pOV3cydWxvDQp1Njl0V1Zjd1pJcEQwU2RwMTdBWVB5WFliZ3R5VG41aldONldnSk83VTRxVHA2MkNOSUJJa1JnQzNJR2ROK1pmDQpVQm9lNVRiNHkyMEZKUWVxQXYwbWhuajdxOXRMZEdCSHYzcTQ0MmhmeFNGRTJKVHNHNTRUZlFGeHVjeDlySTlJDQpVdXZRWHo0MW1hcjdOV29QN0YyZ2xqcllIOWt1OEVpTWNYWjZEZ2pEZ1lLL2F5QitpPQTdJL3dad3ZubHFaRkdNDQpGL2c0aHU3OEpsd0N2aG14cDJ5Yld3c1JSRUd0M0RtSDBpbTM5K2dPdzZUTDhnL1FlbXNKeFN6dU9FRlo1S3ZIDQpTcWZjN2FBT1o2ZEZJTnhmQWV1OGJBSTNzTXZiQWp3aW1ZUW41WGRLSjJUSnEwOGdGWkkrODk1Y3ZadGJtZmUxDQpZdjd4bXFFaGlOamxLNGd3azRGVW9mYVlmejJPcnpPMXhWSzBubnZ1Mkx2M0FyWFZzcUhMd3Y0dTFlcUVmcDBBDQpDc05lU01QeFY5T2NzS010UHZ1L0VFQWpNVm44Z1FzR0s4N3dHRHdwemZPVkFKUURpWkRuSzdlSVUxNGxnZVhnDQozQlN5TFM2d1lyR0Q3U3FFK0xrMVU4cHh0UU1WallOVE9lNzBFL0FxV1EreCtZbHQ1SEN6azkxMWQ4
```

```
OVF1V3VuDQpURzFIT1c1TnlJUU9TZnJDV1k2amFaM2NoZWxvQUFtVzNrVmt3YTV1S2x3Vk5kYTNweVhZM2RHRGRwcE9RWkE1DQp
5Zz09DQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tDQo=' --data-raw '{
  "szolgaltatoAdatok": {
    "adoszam": "12345632243",
    "szolgHelyRegisztraciosSzam": "MK22004233"
  },
  "kuldoRendszerAdatok": {
    "tssRendszerNTAKAzonosito": "Jegy.hu",
    "tssRendszerVerzioszam": "1"
  },
  "uzenetAdatok": {
    "uzenetKuldesIdeje": "2022-12-02T17:42:52.517+01:00"
  },
  "jegyek": [
    {
      "ervenyesitesiAzonosito": "9561dad3-673b-4994-83b3-5a672ac2cd08",
      "belepettSzemelyekSzama": 1,
      "ervenyesitesIdopontja": "2022-12-02T17:42:52.517+01:00",
      "felhasznalasModja": "BELEPES",
      "jegyAzonosito": "J-ca062050-1fbe-434c-bebe-533737a4d32b",
      "programazonosito": {
        "tssProgramAzonosito": "P001",
        "utolsoModositasIdeje": "2022-04-06T09:04:44.65+02:00"
      }
    }
  ]
}'
```

6.2.4. Verification

```
curl --location --request POST 'https://tss.ntaktst.hu/api/ellenorzes/statusz' --header 'x-jws-signature:
eyJhbGciOiJSUzI1NiJ9..Q0UNY7W6jN0v03NeoCoMfiCfyofx8VZTiRyr1VDBciSAww22mOVb549IQW6hQ-Zix2OjbTxIilz2gUyznC-
MDv3WGblnysZVbCIEyiCWtJBLygw4Jgecnkwv6QfjJep6mUDuw8AhTy6vLXIDWoAOBwRc1LDmPiALlDgAB5iVgk5cJXKllCVk6hfUMKQ
-z26OHuv9mAonePJeD5nMC1fz38fxnvMFC30LkRcz72ildBJGn_i7qtSL-
t6YIoNYHqWNah9UnVBx27HUcjtqXqf1YxdQh4DeEnZ3upqfb3RhlnIjuWChlnk3asNGHO05zXbGDJf2PEMzDKZx1N5vng3K6zYbj64cNY
Ujia2WEbXedSoe5GKe_SoBOq6mn-f4zEEYWORSuxo9uF6e71TEw5p2z5U41EYskt4Lc2uRzX-
7MLAuWpPxsKxsonCr7GtwrDAJc_RuB6TtXEpxyAtpHVmL6Q8FVDXoHH1wmYO3OQzFkCam54VMuLRurYCv4QYBX7sDmGHfqPs7_w
yUL8q5yp-
xbxftB7VkpcJFPTkyRlsD3fyUQqD929d7xHl99MM5r6GscRqUwdU6AbbKcXEukNjzr6hGGwjLIVJhZ4FSOd2QqCPnZkydNg1NxHEiUrx9
Fx7Li-s8dsQ32o64yxSHRwmnbVVKbiXGFUW4CpvEHVCNyNg' --header 'x-certificate:
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tDQpNSUlIZlRDQ0JXV2dBd0lCQWdJUkFVZ2MrSnRGa2pKR3NnYjlqamlHaHpnd0RRWUpL
b1pJaHZjTkFRRUxCUUF3DQpnWTB4TWpBd0JnTlZCQU1NS1ZSRlUxcFVJRTVVUVVzdFEwRWdRMnhwWlc1MElFRjFkR2dnVTJsbmJtb
HVaeUJEDQpRU0F5TURJeU1Sc3dHUVlEVlFRTERCSlVSVk5hVkNCT1ZFRkxMVU5CSURJd01qSXhHakFZQmdOVkJBb01FVlJGDQpVMX
BVSUVSQlZGSkJTeUJMWm5RdU1SRXdEd1lEVlFRSERBaENkV1JoY0dWemRERUxNQWtHQTFVRUJoTUNTRlV3DQpIaGNOTWpJeE1q
QXlNVFl3TmpJd1doY05NaIF4TWpBeE1UWXdOakl3V2pDQm5URUxNQWtHQTFVRUJoTUNTRlV4DQpHREFXQmdOVkJBY01EMEoxW
kdGd1pYTjBJQzBnU1VsSkxqRU5NQXNHQTFVRUVRd0VNVEF6TkRFUk1BOEdBMVVFDQpZUXdJTVRJek5EVTJNekl4R2pBWUJnTlZCQ
W9NRVZSVFV5QlNReUJOU3pJeU1EQTBNak16TVJFd0R3WURWUVFEDQpEQWd5TWpBd05ESXpNekVSTUE4R0ExVUVDd3dJVXpRN
FNrRkdPRWN4RURBT0JnTlZCQVVUQjBwbFoza3VhSFV3DQpnZ0lpTUEwR0NTcUdTSWIzRFFFQkFRVUFBNElDRHdBd2dnSUtBb0lDQV
FDcEVMTGp0eDBERGtOdmUzd1k5TTJvDQo0SjdMT0c1bkVyY1dwSkF4ZjZ4aThxVHlTRW11NTBYQmlUa0FLNUNWZVRBOThvNEEw
WUs5OCtuMzB3ZHFFVkM3DQpvcTVkZ3VVakt6M1RwRlpKTWR3Qzc1NGNSVXYyTjJwRk0ySzJ2bFdtd0pjcVBLMlZVanNMR0ZJd3EvN
HYwMWZTDQpuV3pnNGUxZHJjOVZZZVN4UnA0STZBUm1PQnlaQjNwbDdXYmZiQ1YrZmU1TFFYTjJ3QVhCV2ZKb2xrS3lpaFRnDQpx
OEoxODkvOXBHckRCbWs0dFVUTkNiaXVpZklOM3NWeGcrUHZUTzZVN1ExTVM2MFd3VW5JL1NWajJnUm9TTTY1DQpsUjRYU25vSH
BLcDB5SCtuWGIvdnAwZTc5T04zWnVYWnB3aFlidGhXWllNcFlXN3JNcUVZZW43S0MzZjh1QmlEDQo5QlVZZWoxa3M0YWc1V0xXS0J
hWW9PQkRucHJDWlZaSFRHa0xrTnQ3UHRwVVlxOWdIVW8rRU95YkZmNENvSk1JDQpOZ1VWNERxd2ZUNDQrWkROSmF5M3F6Tlp
iNEQvemtJb1N3bzRFRUNyS1N2Vk1UVGo0Ry9lb2NmQk1DRWI0S1BSDQpONkpQT2tldzRwbGdYaDR5QllkSy82amdHZEdRWFNORE
VGVWorZjdTMXJ0SURrcnpjeFhmL3pYaEdWMzl5cDVSDQpDblNTZTROaThvYzNjUmR5dVZGOHZaalVCSzNURERCbXlZdzZvNFUyWGR
Ka2F5ZXRVRCtwamhnRUUrc1ViNHM5DQpzL3doWnJheW5WaW5KYWp2WGRzVjZCblVhMGMwL3ZlemZFenQ2Y2dERTUrSGh1WlB
xaVRpY2dXVUM1NlBQODR5DQpaYUVJQzNPSUN1RTNoSWRkVTh4SWZRSURBUUFCbzRJQnhEQ0NBY0F3Q1FZRFZSMFRCQUl3QUR
CckJnTlZIUjhFDQpaREJpTUdDZ1hxQmNobHBvZEhSd09pOHZkR1Z6ZW5RdWJuUmhhMk5oTG1oMUwyNTBZV3RqWVdOc2FXVnVkR
0YxDQpkR2h6YVdkdWFXNW5ZMkV5TURJeUwyTnliQzl1ZEdGclkyRmpiR2xsYm5SaGRYUm9jMmxuYm1sdVoyTmhNakF5DQpNaTVq
Y213d2djUUdDQ3NHQVFVRkJ3RUJCSUczTUlHME1Hc0dDQ3NHQVFVRkJ6QUNobDlvZEhSd09pOHZkR1Z6DQplblF1Ym5SaGEyTmh
```

```
MbWgxTDI1MFlXdGpZV05zYVdWdWRHRjFkR2h6YVdkdWFXNW5ZMkV5TURJeUwyTmhhWE56DQpkV1Z5TDI1MFlXdGpZV05zYVd
WdWRHRjFkR2h6YVdkdWFXNW5ZMkV5TURJeUxtTmxjakJGQmdnckJnRUZCUWN3DQpBWVk1YUhSMGNEb3ZMM1JsYzNwMExtNT
BZV3RqWVM1b2RTOXVkR0ZyWTJGamJHbGxiblJoZFhSb2MybG5ibWx1DQpaMk5oTWpBeU1pOXZZM053TUE0R0ExVWREd0VCL3d
RRUF3SUY0REFhQmdOVkhTQUVFekFSTUE4R0RTc0dBUVFCDQpnOE1hQVFNQkF3SXdFd1lEVlIwbEJBd3dDZ1lJS3dZQkJRVUhBd0l3S
HdZRFZSMGpCQmd3Rm9BVVRPVXFmVWQ5DQp1M1N1Z09rZHV6eCtQVDBJVExFd0hRWURWUjBPQkJZRUZQNnZYRFc0ZXJBVGhkT
mlNaEdRVHlxNWljcEtNQTBHDQpDU3FHU0liM0RRRUJDd1VBQTRJQ0FRQllkRzU2c20wR0FncU8xT3hHRFhYT3ZiWnYwbkdMcWFhR
VY4T0h3L3pPDQo3SVlLZWVtWmRicURONG1rdFRsSzkyLytaN2dsTGV3MzBrbExSVklDRGRxWmpRR0ZXVDhnMkR3d2pOV3cydWxvD
Qp1Njl0V1Zjd1pJcEQwU2RwMTdBWVB5WFliZ3R5VG41aldONldnSk83VTRxVHA2MkNOSUJJa1JnQzNJR2ROK1pmDQpVQm9lNVRiN
HkyMEZKUWVxQXYwbWhuajdxOXRMZEdCSHYzcTQ0MmhmeFNGRTJKVHNHNTRUZlFGeHVjeDlySTlJDQpVdXZRWHo0MW1hcjdOV
29QN0YyZ2xqcllIOWt1OEVpTWNYWjZEZ2pEZ1lLL2F5QitPQTdJL3dad3ZubHFaRkdNDQpGL2c0aHU3OEpsd0N2aG14cDJ5Yld3c1JSR
Ud0M0RtSDBpbTM5K2dPdzZUTDhnL1FlbXNKeFN6dU9FRlo1S3ZIDQpTcWZjN2FBT1o2ZEZJTnhmQWV1OGJBSTNzTXZiQWp3aW1Z
UW41WGRLSjJUSnEwOGdGWkkrODk1Y3ZadGJtZmUxDQpZdjd4bXFFaGlOamxLNGd3azRGVW9mYVlmejJPcnpPMXhWSzBubnZ1Mk
x2M0FyWFZzcUhMd3Y0dTFlcUVmcDBBDQpDc05lU01QeFY5T2NzS010UHZ1L0VFQWpNVm44Z1FzR0s4N3dHRHdwemZPVkFKUUR
pWkRuSzdISVUxNGxnZVhnDQozQlN5TFM2d1lyR0Q3U3FFK0xrMVU4cHh0UU1WallOVE9lNzBFL0FxV1EreCtZbHQ1SEN6azkxMWQ4
OVF1V3VuDQpURzFIT1c1TnlJUU9TZnJDV1k2amFaM2NoZWxvQUFtVzNrVmt3YTV1S2x3Vk5kYTNweVhZM2RHRGRwcE9RWkE1DQp
5Zz09DQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tDQo=' --header 'Content-Type: application/json' --data-raw '{
  "feldolgozasAzonositok": [
    "402d3781-80f0-435b-b784-ba7870211b1d","f147f8d8-3650-4196-be94-43d4401de379"
  ],
  "szolgaltatoAdatok": {
    "adoszam": "12345632243",
    "szolgHelyRegisztraciosSzam": "MK22004233"
  },
  "kuldoRendszerAdatok": {
    "tssRendszerNTAKAzonosito": "Jegy.hu",
    "tssRendszerVerzioszam": "1"
  },
  "uzenetAdatok": {
    "uzenetKuldesIdeje": "2022-12-02T17:43:09.111+01:00"
  }
}'
```

6.3. Country codes

Country name	Value to send in the message
Afghanistan	AF
Åland Islands	AX
Albania	AL
Algeria	DZ
United States Minor Outlying Islands	UM
United States of America	US
American Samoa	AS
Virgin Islands (U.S.)	VI
Andorra	AD
Angola	AO
Anguilla	AI
Antarctica	AQ
Antigua and Barbuda	AG
Argentina	AR
Aruba	AW
Australia	AU
Austria	AT
Azerbaijan	AZ
Bahamas	BS
Bahrain	BH
Bangladesh	BD
Barbados	BB
Belgium	BE
Belize	BZ
Benin	BJ
Bermuda	BM
Bhutan	BT
Guinea-Bissau	GW
Bolivia	BO
Bosnia and Herzegovina	BA
Botswana	BW
Bouvet Island	BV
Brazil	BR
British Indian Ocean Territory	IO
Virgin Islands (British)	VG
Brunei Darussalam	BN
Bulgaria	BG
Burkina Faso	BF
Burundi	BI
Chile	CL
Cyprus	CY
Comoros	KM
Cook Islands	CK
Costa Rica	CR
Curaçao	CW

Country name	Value to send in the message
Chad	TD
Czech Republic	CZ
Denmark	DK
South Africa	ZA
South Georgia and the South Sandwich Island	GS
South Sudan	SS
South Korea	KR
Dominica	DM
Dominican Republic	DO
Djibouti	DJ
Ecuador	EC
Equatorial Guinea	GQ
United Arab Emirates	AE
United Kingdom of Great Britain and Northern Ireland	GB
Egypt	EG
Cote D'Ivoire	CI
El Salvador	SV
Eritrea	ER
Northern Mariana Islands	MP
North Korea	KP
Estonia	EE
Ethiopia	ET
Falkland Islands (Malvinas)	FK
Belarus	BY
Faroe Islands	FO
Fiji	FJ
Finland	FI
French Southern Territories	TF
French Guiana	GF
France	FR
French Polynesia	PF
Philippines	PH
Gabon	GA
Gambia	GM
Ghana	GH
Gibraltar	GI
Greece	GR
Grenada	GD
Greenland	GL
Georgia	GE
Guadeloupe	GP
Guam	GU
Guatemala	GT
Guernsey	GG
Guinea	GN
Guyana	GY
Haiti	HT
Heard Island and McDonald Islands	HM

Country name	Value to send in the message
Netherlands	NL
Honduras	HN
Hong Kong	HK
Croatia	HR
India	IN
Indonesia	ID
Iraq	IQ
Iran	IR
Ireland	IE
Iceland	IS
Israel	IL
Jamaica	JM
Japan	JP
Yemen	YE
Jersey	JE
Jordan	JO
Cayman Islands	KY
Cambodia	KH
Cameroon	CM
Canada	CA
Christmas Island	CX
Bonaire, Sint Eustatius and Saba	BQ
Qatar	QA
Kazakhstan	KZ
Timor-Leste	TL
Kenya	KE
China	CN
Kyrgyzstan	KG
Kiribati	KI
Cocos (Keeling) Islands	CC
Colombia	CO
Congo (Democratic Republic of the)	CD
Congo (Republic of the)	CG
Kosovo	XK
Central African Republic	CF
Cuba	CU
Kuwait	KW
Lao People's Democratic Republic	LA
Poland	PL
Lesotho	LS
Latvia	LV
Lebanon	LB
Liberia	LR
Libya	LY
Liechtenstein	LI
Lithuania	LT
Luxembourg	LU
North Macedonia	MK

Country name	Value to send in the message
Madagascar	MG
Hungary	HU
Macao	MO
Malaysia	MY
Malawi	MW
Maldives	MV
Mali	ML
Malta	MT
Isle of Man	IM
Morocco	MA
Marshall Islands	MH
Martinique	MQ
Mauritania	MR
Mauritius	MU
Mayotte	YT
Mexico	MX
Myanmar	MM
Micronesia, Federated States of	FM
Moldova, Republic of	MD
Monaco	MC
Mongolia	MN
Montenegro	ME
Montserrat	MS
Mozambique	MZ
Namibia	NA
Nauru	NR
Germany	DE
Nepal	NP
Nicaragua	NI
Niger	NE
Nigeria	NG
Niue	NU
Norfolk Island	NF
Norway	NO
Western Sahara	EH
Italy	IT
Oman	OM
Russian Federation	RU
Armenia	AM
Pakistan	PK
Palau	PW
Palestine, State of	PS
Panama	PA
Papua New Guinea	PG
Paraguay	PY
Peru	PE
Pitcairn	PN
Portugal	PT

Country name	Value to send in the message
Puerto Rico	PR
Reunion	RE
Romania	RO
Rwanda	RW
Saint Barthélemy	BL
Saint Kitts and Nevis	KN
Saint Lucia	LC
Saint Martin (French part)	MF
Saint Pierre and Miquelon	PM
Saint Vincent and the Grenadines	VC
Solomon Islands	SB
San Marino	SM
Sao Tome and Principe	ST
Seychelles	SC
Sierra Leone	SL
Sint Maarten (Dutch part)	SX
Spain	ES
Svalbard and Jan Mayen	SJ
Sri Lanka	LK
Suriname	SR
Switzerland	CH
Sweden	SE
Samoa	WS
Saudi Arabia	SA
Senegal	SN
Saint Helena	SH
Serbia	RS
Serbia and Montenegro	CS
Singapore	SG
Syrian Arab Republic	SY
Slovakia	SK
Slovenia	SI
Somalia	SO
Sudan	SD
Eswatini	SZ
Tajikistan	TJ
Taiwan, Province of China	TW
Tanzania, United Republic of	TZ
Thailand	TH
Togo	TG
Tokelau	TK
Tonga	TO
Turkey	TR
Trinidad and Tobago	TT
Tunisia	TN
Turks and Caicos Islands	TC
Tuvalu	TV
Turkmenistan	TM

Country name	Value to send in the message
Uganda	UG
New Caledonia	NC
New Zealand	NZ
Ukraine	UA
Uruguay	UY
Uzbekistan	UZ
Vanuatu	VU
Holy See (Vatican City State)	VA
Venezuela	VE
Vietnam	VN
Wallis and Futuna	WF
Zambia	ZM
Zimbabwe	ZW
Cape Verde	CV

6.4. Programme categories

Programme main category	Value to send in the message	Programme subcategory	Value to send in the message	Where applicable
Public baths	KOZFURDO	Spa Programme, Beach Programme, Swimming Pool Programme, Health Services, Wellness Programme	FURDO_PROGRAM, STRAND_PROGRAM, USZODAI_PROGRAM, WELLNESS_PROGRAM, EGESZSEGUGYI_SZOLGALTATAS	Adventure Spa, Indoor Adventure Spa, Medicinal Spa, Aquapark, Beach, Thermal Spa, Swimming Pool, Wellness Spa, Complex Spa, Day Spa, Mofetta, Medicinal Cave
Natural bathing site	TERMESZETES_FURDO	Outdoor beach programme	SZABADVIZI_STRAND_PROGRAM	Outdoor beach
Museum, castle, palace	MUZEUM_VAR_KASTELY	Exhibition, Museum education, Guided Tour, Educational programme, Religious facility visit, A visit to a palace, Castle visit, A visit to other monuments	KIALLITAS, MUZEUMPEDAGOGIA, TARLATVEZETES, EGYHAZI_LETESITMENY_LATOGATASA, KASTELYLATOGATAS, VARLATOGATAS, EGYEB_MUEMLEK_LATOGATASA, OKTATASI_PROGRAM	Church collection, Memorial museum, Local historical collection, Exhibition site/Presentation site, Visitor centre, Heritage memorial site, National historical park, Outdoor museum, Specialist collection, Specialist museum, Country house, Country museum, Agricultural heritage, Church building, Industrial heritage, Castle, Mansion, Folk architectural heritage, Palace, Ruin, Mausoleum, Historical garden, Fortress, Secular public building, Basilica/Minor, World heritage, Other

Programme main category	Value to send in the message	Programme subcategory	Value to send in the message	Where applicable
Tourist transport infrastructure transport facility	TURISZTIKAI_KOZLEKEDES	Lake cruises, River cruises, Hop on hop off program	TAVI_HAJOZAS, FOLYAMI_HAJOZAS, HOP_ON_HOP_OFF	Lake cruises, River cruises, Hop on hop off program, E-mobility
Adventure Park	KALANDPARK	Adventure Park Programme, Experience Centre Programme	KALANDPARKI_PROGRAM, ELMENYKOZPONTI_PROGRAM	Adventure Park, Experience Centre
Zoo	ALLATKERT	Zoo programme, Wildlife park programme	ALLATKERTI_PROGRAM, VADASPARKI_PROGRAM	Zoo, Wildlife park
Amusement park	VIDAMPARK	Amusement park programme	VIDAMPARKI_PROGRAM	Amusement park
Protected natural area showcase	VEDETT_TERMESZETI_TERULET	National Park visit, Cave visit, Guided Tour, Educational Programme, Park visit, Theme Demo Park visit, Tours/Trips	NEMZETI_PARK_LATOGATAS, BARLANGLATOGATAS, TARLATVEZETES, OKTATASI_PROGRAM, PARKLATOGATAS, TEMATIKUS_BEMUTATOHELY_LATOGATAS, TURAK_KIRANDULASOK	National park, Cave, Archaeological site, Fossil, Reservation, Reception, visitor and training centre, Arboretum, Botanic garden, Thematic demo park, Tours/Trips
Music and dance event	ZENES_TANCOS_RENDEZVENY	Cultural event, Concert, Festival	KULTURALIS_RENDEZVENY, KONCERT, FESZTIVAL	Festival, Concert, Performance, Ceremonial event, Family and kids' concert, Musical event, Other cultural event (music events for more than 1,000 persons, with standing places)