

*Labelling IS. Monitoring of medicinal product movement (MMM).  
Interface level exchange protocol.*

*Revision 2.6*

## Change history

Change date	Revision	Change description
19.10.2017	2.1	<p>The following methods were added:</p> <ul style="list-style-type: none"> <li>receipt of information from the register of Manufacturing Licenses (for personal Licenses);</li> <li>receipt of information from the register of licenses for pharmaceutical activities (for personal Licenses);</li> <li>receipt of information from the register;</li> <li>receipt of a list of proprietary places of business; receipt of a list of proprietary consignment warehouses.</li> </ul>
27.10.2017	2.2	<p>In the following methods <i>user</i> was replaced with <i>users</i> in the context path:</p> <ul style="list-style-type: none"> <li>● Method of deleting accounting system users</li> <li>● Method of changing user electronic signature (for residents)</li> <li>● Method of changing user password (for non-residents)</li> <li>● Method of searching by registered users</li> </ul> <p>In the following methods the http method was changed from POST to PUT:</p> <ul style="list-style-type: none"> <li>● Method of changing user electronic signature (for residents)</li> <li>● Method of changing user password (for non-residents)</li> </ul> <p>An outgoing data block was added for the following method:</p> <ul style="list-style-type: none"> <li>● Method of receiving user information</li> </ul> <p>The description of the password attribute was changed in the method of changing the user password (for non-residents): <i>Password hash</i> was replaced with <i>Password</i></p> <p>Operation completion codes were replaced with http codes in replies to http requests.</p> <p>The <i>Operation status</i> attribute in JSON format was deleted in all methods with output data containing the attribute.</p> <p>The list of accounting system user rights (RightsEnum) was changed for the purpose of account management: the following rights</p> <ul style="list-style-type: none"> <li>● registration_resident_user</li> <li>● registration_nonresident_user</li> <li>● registration_account_system</li> <li>● set_rights</li> <li>● delete_user</li> </ul> <p>replaced with:</p> <ul style="list-style-type: none"> <li>● manage_accounts</li> <li>● view_accounts</li> </ul>
28.10.2017	2.3	Correction of typing errors
17.11.2017	2.4	<p>1) In all methods, the <i>token</i> parameter was deleted from the list of parameters transferred in the request body. Authorization is completed using the <i>Authorization</i> http header</p> <p>2) Information on methods requiring/not requiring authorization was added to section 3.2. User authentication and authorization</p> <p>3) in the following http methods the method was changed from GET to POST:</p> <ul style="list-style-type: none"> <li>● Receipt of a text address by FIAS identifiers</li> </ul>

		<p>4) The path was changed to &lt;endpoint&gt;/&lt;version&gt;/reestr/egrip in the “Receipt of USRIE entry data” method</p> <p>5) Method of receiving a document by identifier was added</p> <p>6) the following attribute in the JSON format was added to the format of the AccountSystem object:</p> <ul style="list-style-type: none"> <li>● account_system_id</li> </ul> <p>7) in the format of the User object for the groups field the JSON data type was changed to Array of Strings – a list of group names</p> <p>8) The registers were supplemented with a method of receiving information from the ESKLP register</p>
27.11.2017	2.5	A section with examples of authorization using test participants was added.
30.11.2017	2.6	<p>A section with a description of the format of a signature used in the methods of sending documents and authorization of resident users:</p> <ul style="list-style-type: none"> <li>● Format of electronic signature</li> </ul> <p>A description of the certificate identifier for the method was added</p> <ul style="list-style-type: none"> <li>● Method of receiving an authentication code</li> </ul> <p>Output information was updated (user_id was added) fro the methods</p> <ul style="list-style-type: none"> <li>● Method of accounting system users registration (for residents of a country)</li> <li>● Method of accounting system users registration (for non-residents of a country)</li> </ul>

# Contents

Access to methods of the “Labelling IS. MMM” interface .....	7
1. General data structures, restrictions and operation completion codes .....	7
1.1. Restrictions .....	7
1.2. Operation completion code .....	9
1.3. Format of electronic signature .....	9
1.4. The list of accounting system users’ rights (RightsEnum) .....	10
1.5. The list of document statuses (DocStatusEnum) .....	11
1.6. List of authentication types (AuthType) .....	12
1.7. The format of Document object .....	12
1.8. The format of DocFilter object .....	13
1.9. The format of Group object .....	14
1.10. Format of the GroupInfo object .....	14
1.11. The format of User object .....	14
1.12. The format of UserInfo object .....	15
1.13. The format of AccountSystem object .....	15
1.14. Format of the RightsInfo object .....	16
1.15. The format of ADDRESS object .....	16
2. Uploading of documents .....	16
2.2. Sending of a large document .....	17
2.3. Completion of the document sending process .....	18
2.4. Obtain information on size restrictions for small documents .....	19
2.5. Cancellation of document sending .....	20
2.6. Receipt of a list of outgoing documents .....	20
2.7. Receipt of an incoming document list .....	21
2.8. Receipt of document meta-data .....	22
2.9. Receipt of a document by identifier .....	22
2.10. Receipt of a list of documents by request identifier .....	23
3. Users and accounting systems .....	24
3.1. Registration of an accounting system and accounting system users .....	24
3.1.1. Method of registering an accounting system .....	24
3.1.2. Method of accounting system users registration (for residents of a country) .....	25
3.1.3. Method of accounting system users registration (for non-residents of a country) .....	26
3.1.4. Method of receiving user information .....	27
3.1.5. Method of receiving information on the current user .....	27
3.1.6. Method of receiving AS information .....	28
3.2. User authentication and authorization .....	28
3.2.1. Method of receiving an authentication code .....	28

3.2.2.	Method of receiving a session key .....	30
3.2.3.	Logout method .....	30
3.3.	Deletion of accounting systems and accounting system users.....	31
3.3.1.	Method of deleting accounting system users .....	31
3.3.2.	Method of deleting an accounting system.....	31
3.4.	Change of user electronic signature (for residents).....	31
3.4.1.	Method of changing user electronic signature (for residents) .....	31
3.5.	Change of user password (for non-residents).....	32
3.5.1.	Method of changing user password (for non-residents) .....	32
3.6.	User right groups.....	32
3.6.1.	Method of receiving information on right variations .....	32
3.6.2.	Method of creating a user right group.....	33
3.6.3.	Method of receiving information on a user right group .....	33
3.6.4.	Method of receiving group user information .....	34
3.6.5.	Method of changing a user right group .....	35
3.6.6.	Method of deleting a user rights group .....	35
3.6.7.	Method of adding a user to a user right group .....	36
3.6.8.	Method of deleting a user from a user right group .....	36
3.6.9.	Method of receiving a list of user rights groups .....	37
3.7.	Search by registered users .....	37
3.7.1.	Method of searching by registered users .....	38
3.8.	Search by AS .....	38
3.8.1.	Method of search by AS .....	38
4.	Receipt of information from registers.....	39
4.1.	Allows to obtain information from USRLE register .....	39
4.1.1.	Receipt of USRLE entry data.....	39
4.2.	Allows to obtain information from USRIE register .....	40
4.2.1.	Receipt of USRIE (Uniform State Register of Individual Entrepreneurs) entry data .....	40
4.3.	Allows to obtain information from register of certified branches and representative offices.....	41
4.3.1.	Receipt of a RAPF (Register of Accredited Branches and Representative Offices of Foreign Legal Entities) register entry .....	41
4.4.	Receipt of information from the register of taxes payable.....	41
4.4.1.	Receipt of debt information .....	41
4.5.	Allows to obtain information from FIAS register .....	43
4.5.1.	Receipt of a FIAS object by address object identifier.....	43
4.5.2.	Receipt of a FIAS object by address object identifier.....	44
4.5.3.	Receipt of a text address by FIAS identifiers.....	45
4.6.	Receipt of information from the register of Manufacturing Licenses .....	45
4.6.1.	Receipt of information on Manufacturing Licenses.....	45
4.7.	Receipt of information from the register of licenses for pharmaceutical activities .....	46

4.7.1.	Receipt of information on licenses for pharmaceutical activities.....	46
4.8.	Receipt of information on places of business .....	47
4.8.1.	Receipt of information on places of business .....	47
4.8.2.	Receipt of information on a specific place of business .....	48
4.9.	Receipt of information on consignment warehouses .....	49
4.9.1.	Receipt of information on consignment warehouses .....	49
4.9.2.	Receipt of information on a specific consignment warehouse.....	49
4.10.	Allows to obtain information from ESKLP register.....	50
4.10.1.	Filtering by register .....	50
5.	Examples of authorization on test participants. ....	56
5.1.	Test data of Participant No.1 .....	56
5.2.	Example of authorization for test participant No.1 .....	57
5.3.	Test data of Participant No.2 .....	58
5.4.	Example of authorization for test participant No.2 .....	59

## Access to methods of the “Labelling IS. MMM” interface

All methods of “Labelling IS. MMM” interface except for “Method of receiving an authentication code” (3.2.1), “Method of receiving a session code” (3.2.2) and “Method of receiving information on size restriction for small documents” (2.4) require a specific header:

Authorization: token <token\_id>

where token\_id is a session key received using “Method of receiving a session key” (3.2.2)

## 1. General data structures, restrictions and operation completion codes

### 1.1. Restrictions

There are the following common restrictions for data field formats of input requests:

Restriction	Designation	JSON data type	Format
Identifier of circulation subject in “Product labelling” IS	SysID	String	36 symbols Template (as per RFC4122): ([a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12})
Identifier of a circulation subject - account subject place of business in “Product labelling” IS	BranchID	String	14 symbols Template: [0-9]{14}
Identification number to identify goods (GTIN)	GTIN	String	14 symbols Template: [0-9]{14}
Global unique identifier	GUID	String	36 symbols Template (as per RFC4122): ([a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12})
Data and time format with the obligatory indication of time zone (or in UTC format)	Datetime	String	Template (as per RFC 3339): /^\d\d\d\d-[0-1]\d-[0-3]\d[t\s][0-2]\d:[0-5]\d:[0-5]\d(?:\.\d+)?(?:z [+-]\d\d:\d\d)\$/i

Date in the format YYYY-MM-DD	Date	String	10 symbols Template (as per RFC 3339): /^\d\d\d\d-(\d\d)-(\d\d)\$/
Production batch number	BatchNumber	String	1-20 symbols Template: \S+(*\S+)*
Code from EEU FEACN (the first 4 symbols of the code)	TNVED4	String	4 symbols Template: [0-9]{4}
Identifier No. of a taxpayer - physical person	INNFL	String	12 symbols Template: ([0-9]{1}[1-9]{1} [1-9]{1}[0-9]{1})[0-9]{10}
Identifier No. of a taxpayer - legal entity	INNUL	String	10 symbols Template: ([0-9]{1}[1-9]{1} [1-9]{1}[0-9]{1})[0-9]{8}
Identifier No. of a taxpayer - foreign manufacturer	ITIN	String	1-50 symbols Template: .{1,50}
Registration reason code (RRC)	KPP	String	9 symbols Template: .([0-9]{1}[1-9]{1} [1-9]{1}[0-9]{1})[0-9]{7}
Document No. (no more than 200 symbols)	DocNumber	String	1-200 symbols Template: .{1,200}
Document name (no more than 200 symbols)	DocName	String	1-200 symbols Template: .{1,200}
Price	Price	Number	Value in kopecks



Customs code	FTSCC	String	2, 5 or 8 symbols
Code of a decision made by customs	FTSDecision	String	2 symbols

### 1.2. Operation completion code

Methods of services return http codes as REST services as a result of completing an operation:

Code value	Code definition	Description
200	OK	Successful operation completion
403	FORBIDDEN	Insufficient right to execute the operation
400	BAD_REQUEST	Error when executing the operation

### 1.3. Format of electronic signature

Data signing is used in various API sections (example: document sending, authorization for residents). The signature should be received in API in base64 (RFC2045) format with the following parameters

Parameter	Value	Note
Standard	PKCS7	
Type of signature	Detached	
The algorithm of signature (OID)	1.2.643.7.1.1.3.3	In crypto-pro it corresponds to GOST_SIGN_2012_512_OID, GOST3410DH_2012_512, GOST3410_2012_512  An older and less reliable algorithm 1.2.643.7.1.1.3.2 (GOST_SIGN_2012_256_OID, GOST3410DH_2012_256, GOST3410_2012_256) will also be supported

Digest (OID)	1.2.643.7.1.1.2.3	<p>In crypto-pro it corresponds to GOST_DIGEST_2012_512_OID, GOST3410DH_2012_512, GOST3410_2012_512.</p> <p>An older and less reliable digest 1.2.643.7.1.1.2.2 will also be supporter (GOST_DIGEST_2012_256_OID, GOST3410DH_2012_256, GOST3410_2012_256)</p>
Public key (OID)	1.2.643.7.1.1.1.2	<p>In crypto-pro it corresponds to GOST_PARAMS_SIG_2012_512_KEY_OID, GOST3410DH_2012_512, GOST3410_2012_512</p> <p>An older and less reliable version 1.2.643.7.1.1.1.1 (GOST_PARAMS_SIG_2012_256_KEY_OID, GOST3410DH_2012_256, GOST3410_2012_256) will also be supported</p>
Cryptographic service provider	Any provider certified in the Russian Federation	Examples: CryptoPro JCP, VipNet CSP, etc.

#### 1.4. The list of accounting system users' rights (RightsEnum)

Right name	Operation type	Right definition
UPLOAD_DOCUMENT	Uploading of a document	Allows to upload documents
OUTCOME_LIST	Data on an outgoing document	Allows to obtain information on outgoing documents
INCOME_LIST	Data on incoming documents	Allows to obtain information on incoming documents
DOWNLOAD_DOCUMENT	Receiving a link to the document	Allows to obtain a link to the document with the use of document identifier.

MANAGE_ACCOUNTS	Account management	Allows to manage account systems, users and user right groups
VIEW_ACCOUNTS	Registration of an accounting system user (for non-residents)	Allows to browse account systems, users and user right groups
REESTR_ALL	Access to registers	Allows to receive access to all registers
REESTR_FEDERAL_SUBJECT	Register of RF territorial entities	Receipt of information from the register of RF territorial entities
REESTR_EGRUL	USRLE (Uniform State Register of Legal Entities) register	Allows to obtain information from USRLE register
REESTR_EGRIP	USRIE (Uniform State Register of Individual Entrepreneurs) register	Allows to obtain information from USRIE register
REESTR_REFP	Register of certified branches and representative offices	Allows to obtain information from register of certified branches and representative offices
REESTR_DUES	Register of taxes payable	Receipt of information from the register of taxes payable
REESTR_PROD_LICENSES	Register of Manufacturing Licenses	Receipt of information from the register of Manufacturing Licenses
REESTR_PHARM_LICENSES	Register of licenses for pharmaceutical activities	Receipt of information from the register of licenses for pharmaceutical activities
REESTR_ESKLP	ESKLP (Medicine Compendium) register	Allows to obtain information from ESKLP register
REESTR_GS1	GS1 register	Allows to obtain information from GS1 register
REESTR_FIAS	FIAS (Federal Information Address System) register	Allows to obtain information from FIAS register

### 1.5. The list of document statuses (DocStatusEnum)

Status name	Operation type	Status description
-------------	----------------	--------------------

UPLOADING_DOCUMENT	Uploading of a document	Document is being uploaded
PROCESSING_DOCUMENT	Primary processing of a document	Document is received and processed by a converter
CORE_PROCESSING_DOCUMENT	Processing of a document by the system	Document is processed by a converter and accepted for processing by the system
CORE_PROCESSED_DOCUMENT	Preparation of a reply	Document has been processed by the system, and the converter is preparing a reply
PROCESSED_DOCUMENT	The document has been processed, and a reply has been prepared	Document was processed by a converter and is ready for uploading

#### 1.6. List of authentication types (AuthType)

Type name	Status description
PASSWORD	Authentication with a password
SIGNED_CODE	Authentication with a signed one-time code

#### 1.7. The format of Document object

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
request_id	Unique request identifier	String	GUID	1	
document_id	Unique document identifier	String	GUID	1	
date	Document receipt date	String	Date	1	
sender	Sender of a document	String		1	

sys_id	Identifier of circulation subject in "Product labelling" IS	String	SysID	1	
doc_type	Document type	Number	ACTION_ID	1	Corresponds to xsd scheme No.
doc_status	Document status	String	DocStatusEnum	1	

### 1.8. The format of DocFilter object

filter object contains information to filter the list of documents:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
start_date	Starting date of filtration period	String	Date	0	
end_date	Expiration date of filtration period	String	Date	0	
document_id	Unique document identifier	String	GUID	0	
request_id	Unique request identifier	String	GUID	0	
sys_id	Identifier of circulation subject in "Product labelling" IS	String	SysID	0	
doc_type	Document type	Number		0	Corresponds to xsd scheme No.
doc_status	Document status	String	DocStatusEnum	0	

### 1.9. The format of Group object

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_name	Group name	String		1	
rights	Rights of users who are the members of this group	Array of Strings	RightsEnum	1	
group_id	Unique group identifier	String	GUID	1	
users	Users who are the members of this group	Array of Objects	User	1	

### 1.10. Format of the GroupInfo object

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_name	Group name	String		0	
rights	Rights of users who are the members of this group	Array of Strings	RightsEnum	0	

### 1.11. The format of User object

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
user_id	Unique user identifier	String	GUID	1	

first_name	User's name	String		1	
last_name	User's surname	String		1	
middle_name	User's patronymic name	String		0	
groups	Groups of rights which the user is a member of	Array of Strings	Group name	1	

### 1.12. The format of UserInfo object

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
first_name	User's name	String		0	
last_name	User's surname	String		0	
middle_name	User's patronymic name	String		0	

### 1.13. The format of AccountSystem object

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
account_system_id	Unique accounting system identifier				
name	Accounting System (AS) name	String		1	
api_key	API security key	String	GUID	1	Used as client_id when registering AS and receiving an authentication code

### 1.14. Format of the RightsInfo object

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
right	Pseudonym of a right in the system	String	RightsEnum	1	
description	Right description	String		1	

### 1.15. The format of ADDRESS object

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
aoguid	Global unique identifier of an address object (FIAS)	String		1	
houseguid	Global unique identifier of a house (FIAS)	String		1	
address_description	Text address of the object	String		1	

## 2. Uploading of documents

### 2.1. Sending of a document

*Document sending method*

POST <endpoint>/<version>/documents/send

*Parameters of document sending method*

Input information is data on a document



Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
doc_type	Document type	Number		1	Corresponds to xsd scheme No.
document	Document	String	base64	1	Documents encoded with base64 (RFC2045)
sign	Document detached signature	String	base64	1	Ref. section Format of Electronic Signature
request_id	Unique request identifier	String	GUID	1	

### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
document_id	Unique document identifier	String	GUID	0	Upon successful execution of the operation

## 2.2. Sending of a large document

### *Document sending method*

POST <endpoint>/<version>/documents/send\_large

### *Parameters of document sending method*

Input information is data on a document

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
doc_type	Document type	Number		1	Corresponds to xsd scheme No.
sign	Document detached signature	String	base64	1	Ref. section Format of Electronic Signature
hash_sum	Document hash sum	String		1	Hash sum of a document calculated using the <b>SHA-256</b> algorithm
request_id	Unique request identifier	String	GUID	1	

### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
link	Link to WebDAV to upload a document	String		0	Upon successful execution of the operation
document_id	Document Id	String	GUID	0	Upon successful execution of the operation

## 2.3. Completion of the document sending process

*Method of completing the document sending process*

POST <endpoint>/<version>/documents/send\_finished

### *Parameters of document sending method*

Input information is document data:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
document_id	Document Id	String	GUID	1	

### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
request_id	Unique request identifier	String	GUID	0	Upon successful execution of the operation

## 2.4. Obtain information on size restrictions for small documents

*The method of receiving information on size restriction for small documents*

GET <endpoint>/<version>/documents/doc\_size

### *Parameters of document sending method*

No input information is required

### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
doc_size	Document size in bytes	Number		1	Maximum size of a document in bytes which can be sent without WebDAV

## 2.5. Cancellation of document sending

### *Methods of cancelling document sending*

POST <endpoint>/<version>/documents/cancel

### *Parameters of the document sending cancellation method*

Input information is the identifier of the uploaded document:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
document_id	Document identifier	String	GUID	1	
request_id	Unique request identifier	String	GUID	1	

### *Output information*

Based on execution results, the client receives an http operation completion code with an empty reply body.

## 2.6. Receipt of a list of outgoing documents

### *Method of receiving a list of outgoing documents*

POST <endpoint>/<version>/documents/outcome

### *Parameters of receiving a list of outgoing documents*

Input information is data required to filter document list:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
filter	Document list filter	Object	DOC_FILTER	1	
start_from	Index of the first document in delivery	Number		1	

count	The number of document in delivery	Number		1	
-------	------------------------------------	--------	--	---	--

### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
documents	List of documents	Array of objects	Document	1	
total	Overall number of documents upon the request	Number		1	

## 2.7. Receipt of an incoming document list

### *Method of receiving a list of incoming documents*

POST <endpoint>/<version>/documents/income

### *Parameters of receiving a list of incoming documents*

Input information is data required to filter document list:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
filter	Document list filter	Object	DocFilter	1	
start_from	Index of the first document in delivery	Number		1	
count	The number of document in delivery	Number		1	

### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
documents	List of documents	Array of objects	Document	1	
total	Overall number of documents upon the request	Number		1	

## 2.8. Receipt of document meta-data

*Methods of receiving document meta-data*

GET <endpoint>/<version>/documents/{docId}

*Parameters of receiving document meta-data*

Input information is a document identifier:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
document_id	Unique document identifier	String	GUID	1	

*Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
document	Document corresponding to the present id	Object	Document	0	Upon successful execution of the operation

## 2.9. Receipt of a document by identifier

*Method of receiving a document by identifier*

GET <endpoint>/<version>/documents/download/{docId}

*Parameters of receiving a document by identifier*

Input information is a document identifier:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
document_id	Unique document identifier	String	GUID	1	

*Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
link	Link to WebDAV to upload a document	String	PATH	0	Upon successful execution of the operation

## 2.10. Receipt of a list of documents by request identifier

*Method of receiving a document by identifier*

GET <endpoint>/<version>/documents/request/{request\_d}

*Parameters of receiving a document by identifier*

Input information is a document identifier:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
request_id	Unique request identifier	String	GUID	1	

*Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
documents	List of documents	Array of objects	Document	1	
total	Overall number of documents upon the request	Number		1	

### 3. Users and accounting systems

#### 3.1. Registration of an accounting system and accounting system users

##### 3.1.1. Method of registering an accounting system

POST <endpoint>/<version>/registration/accounting\_system

##### *Parameters of the registration method*

Input information is data on an accounting system

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
sys_id	Identifier of circulation subject in "Product labelling" IS	String	SysID	1	
name	Accounting system name	String		1	

##### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
-------------------------------	-----------------------	----------------	--------------	-------------	-------



client_secret	Private key	String	GUID	0	Upon successful execution of the operation
client_id	Client's identifier	String	GUID	0	Upon successful execution of the operation

### 3.1.2. Method of accounting system users registration (for residents of a country)

POST <endpoint>/<version>/registration/user\_resident

#### *Parameters of method of accounting system users registration*

Input information is data on an accounting system

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
sys_id	Identifier of circulation subject in "Product labelling" IS	String	SysID	1	
public_cert	Public certificate of a user	String	base64	1	
first_name	User's name	String		1	
last_name	User's surname	String		1	
middle_name	User's patronymic name	String		0	

#### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
user_id	Unique user identifier	String	GUID	0	Upon successful execution of the operation

### 3.1.3. Method of accounting system users registration (for non-residents of a country)

POST <endpoint>/<version>/registration/user\_nonresident

#### *Parameters of method of accounting system users registration*

Input information is data on an accounting system

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
sys_id	Identifier of circulation subject in "Product labelling" IS	String	SysID	1	
username	Login	String		1	
password	Password	String		1	
first_name	User's name	String		1	
last_name	User's surname	String		1	

#### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
-------------------------------	-----------------------	----------------	--------------	-------------	-------

user_id	Unique user identifier	String	GUID	0	Upon successful execution of the operation
---------	------------------------	--------	------	---	--

### 3.1.4. Method of receiving user information

GET <endpoint>/<version>/users/{user\_id}

#### *Parameters of the method of receiving user information*

Input information is user id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
user_id	Unique user identifier	String	GUID	1	Path parameters (path param)

#### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
user	User	Object	User	0	Upon successful execution of the operation

### 3.1.5. Method of receiving information on the current user

GET <endpoint>/<version>/users/current

#### *Parameters of the method of receiving user information*

Input information is not required

#### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
user	User	Object	User	0	Upon successful execution of the operation

### 3.1.6. Method of receiving AS information

GET <endpoint>/<version>/account\_systems/{account\_system\_id}

*Parameters of the of AS information receipt method*

Input information is AS id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
account_system_id	Unique AS identifier	String	GUID	1	Path parameters (path param)

*Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
account_system	AS	Object	AccountSystem	0	Upon successful execution of the operation

## 3.2. User authentication and authorization

All methods of “Labelling IS. MMM” interface except for “Method of receiving an authentication code” (3.2.1), “Method of receiving a session code” (3.2.2) and “Method of receiving information on size restriction for small documents” (2.4) require a specific header:

Authorization: token <token\_id>

where token\_id is a session key received using the “Method for obtaining a session key” (3.2.2)

### 3.2.1. Method of receiving an authentication code

POST <endpoint>/<version>/auth

*Parameters of the method of receiving an authentication code*

Input information is user data:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
client_id	Client's identifier	String	GUID	1	
client_secret	Client's secret	String	GUID	1	
user_id	User Id	String		1	If identification is conducted for a resident, this field is filled in with a certificate identifier*, and if the user is a non-resident, a login is used
auth_type	Authentication type	String	AuthType	1	{SIGNED_CODE} - electronic digital signature (residents) {PASSWORD} – password (non-residents)

Certificate identifier\* is a certificate's serial number in decimal form.

*Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
code	Code for authentication	String	GUID	0	Upon successful

					completion of an operation
--	--	--	--	--	----------------------------

### 3.2.2. Method of receiving a session key

POST <endpoint>/<version>/token

*Parameters of the method of receiving a session key*

Input information is a password or a signed user code

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
code	Code for authentication	String	GUID	1	
signature	Detached signature of the code for authentication	String	base64	0	Mandatory for type {SIGNED_CODE} Ref. section Format of Electronic Signature
password	User's password	String		0	Obligatory for type {PASSWORD}

#### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
token	AS session key	String	GUID	1	Upon successful execution of the operation
life_time	Life time of a session key	Number		1	In minutes

### 3.2.3. Logout method

GET <endpoint>/<version>/auth/logout

### *Parameters of the method to logging out*

Input information is not required

### *Output information*

Based on execution results, the client receives an http operation completion code with an empty reply body.

## 3.3. Deletion of accounting systems and accounting system users

### 3.3.1. Method of deleting accounting system users

DELETE <endpoint>/<version>/users/{user\_id}

### *Parameters of the method of deleting accounting system users*

Input information is data on an accounting system user:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
user_id	User Id	String	GUID	1	Path param

### *Output information*

Based on execution results, the client receives an http operation completion code with an empty reply body.

### 3.3.2. Method of deleting an accounting system

DELETE <endpoint>/<version>/account\_systems/{account\_system\_id}

### *Parameters of the method of deleting an accounting system*

Input information is accounting system id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
account_system_id	Accounting system Id	String	GUID	1	Path param

### *Output information*

Based on execution results, the client receives an http operation completion code with an empty reply body.

## 3.4. Change of user electronic signature (for residents)

### 3.4.1. Method of changing user electronic signature (for residents)

PUT <endpoint>/<version>/users/{user\_id}/change\_key

### *Parameters of the method of setting rights*

Input information is data on an accounting system user:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
user_id	User Id	String	GUID	1	Path param
public_cert	Public certificate of electronic signature	String	base64	1	

#### *Output information*

Based on execution results, the client receives an http operation completion code with an empty reply body.

### 3.5. Change of user password (for non-residents)

#### 3.5.1. Method of changing user password (for non-residents)

PUT <endpoint>/<version>/users/{user\_id}/change\_password

#### *Parameters of the method of changing the user password (for non-residents)*

Input information is data on an accounting system user:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
user_id	User Id	String	GUID	1	Path param
password	Password	String		1	

#### *Output information*

Based on execution results, the client receives an http operation completion code with an empty reply body.

### 3.6. User right groups

#### 3.6.1. Method of receiving information on right variations

GET <endpoint>/<version>/rights/about

#### *Parameters of the method of receiving information on right variations*

Input information is not required

#### *Output information*

Output information is operation completion status and group id:



Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
rights	Rights and description	Array of Objects	RightsInfo	0	Upon successful execution of the operation

### 3.6.2. Method of creating a user right group

POST <endpoint>/<version>/rights/create\_group

*Parameters of the method of creating user right groups*

Input information is group information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_name	Group name	String		1	
rights	Rights of users belonging to this group	Array of Strings	RightsEnum	1	

*Output information*

Output information is operation completion status and group id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_id	Unique group identifier	String	GUID	0	Upon successful execution of the operation

### 3.6.3. Method of receiving information on a user right group

GET <endpoint>/<version>/rights/{group\_id}

*Parameters of the method of receiving information on a user rights group*

Input information is group id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_id	Unique group identifier	String	GUID	1	Request parameter (Path param)

*Output information*

Output information is operation completion status and information on a group if it s found:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group	Object of Group type	Object	Group	0	Upon successful execution of the operation

3.6.4. Method of receiving group user information

GET <endpoint>/<version>/rights/{group\_id}/users

*Parameters of the method of receiving information on group users*

Input information is group id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_id	Unique group identifier	String	GUID	1	Request parameter (Path param)

*Output information*

Output information is operation completion status and information on a group if it s found:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
-------------------------------	-----------------------	----------------	--------------	-------------	-------

users	Users who are the members of this group	Array of Objects	User	0	Upon successful execution of the operation
-------	---	------------------	------	---	--

### 3.6.5. Method of changing a user right group

PUT <endpoint>/<version>/rights/{group\_id}

*Parameters of the method of changing a user right group*

Input information is group id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_id	Unique group identifier	String	GUID	1	Request parameter (Path param)
group_change	Object of GroupInfo type	Object	GroupInfo	1	

*Output information*

Output information is the operation completion status and updated information on the group, if no errors have occurred:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group	Object of Group type	Object	Group	0	Upon successful execution of the operation

### 3.6.6. Method of deleting a user rights group

DELETE <endpoint>/<version>/rights/{group\_id}

*Parameters of the method of deleting a user rights group*

Input information is group id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
-------------------------------	-----------------------	----------------	--------------	-------------	-------

group_id	Unique group identifier	String	GUID	1	Request parameter (Path param)
----------	-------------------------	--------	------	---	--------------------------------

*Output information*

Based on execution results, the client receives an http operation completion code with an empty reply body.

3.6.7. Method of adding a user to a user right group

POST <endpoint>/<version>/rights/{group\_id}/user\_add

*Parameters of the method of adding a user to a user right group*

Input information is group id and user id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_id	Unique group identifier	String	GUID	1	Request parameter (Path param)
user_id	Unique user identifier	String	GUID	1	

*Output information*

Based on execution results, the client receives an http operation completion code with an empty reply body.

3.6.8. Method of deleting a user from a user right group

DELETE <endpoint>/<version>/rights/{group\_id}/{user\_id}

*Parameters of the method of deleting a user from a user right group*

Input information is group id and user id:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_id	Unique group identifier	String	GUID	1	Request parameter (Path param)

user_id	Unique user identifier	String	GUID	1	Request parameter (Path param)
---------	------------------------	--------	------	---	--------------------------------

#### *Output information*

Based on execution results, the client receives an http operation completion code with an empty reply body.

### 3.6.9. Method of receiving a list of user rights groups

POST <endpoint>/<version>/rights/list

#### *Parameters of the method of receiving a list of user right groups*

Input information is the target group information and the page number in search results:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
group_filter	Object of GroupInfo type	Object	GroupInfo	1	
start_from	Index of the first document in delivery	Number		1	
count	The number of document in delivery	Number		1	

#### *Output information*

Output information is the operation completion status:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
groups	Found groups	Array of Objects	Group	0	Upon successful execution of the operation
total	Overall number of documents upon the request	Number		1	

### 3.7. Search by registered users

### 3.7.1. Method of searching by registered users

POST <endpoint>/<version>/users/find

*Parameters of the method of searching by registered users*

Input information is accounting system user data:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
user_filter	Data for user search	Object	UserInfo	1	
start_from	Index of the first document in delivery	Number		1	
count	The number of document in delivery	Number		1	

#### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
users	Users	Array of objects	User	0	Upon successful execution of the operation
total	Overall number of documents upon the request	Number		1	

### 3.8. Search by AS

#### 3.8.1. Method of search by AS

POST <endpoint>/<version>/account\_systems/find

*Parameters of the method of searching by AS*

Input information is the AS name:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
name	AS name	String		1	
start_from	Index of the first document in delivery	Number		1	
count	The number of document in delivery	Number		1	

#### *Output information*

Based on the results of processing a client receives the following information:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
account_systems	Accounting systems	Array of objects	AccountSystem	0	Upon successful execution of the operation
total	Overall number of documents upon the request	Number		1	

## 4. Receipt of information from registers

4.1. Allows to obtain information from USRLE register

4.1.1. Receipt of USRLE entry data

GET <endpoint>/<version>/reestr/egrul

#### *Method parameters*

Input information is not required

#### *Output information*

Output information contains legal entity data:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
-------------------------------	-----------------------	----------------	--------------	-------------	-------

id	USRLE entry identifier	String		1	
inn	Legal body's INN	String		1	
OGRN	PSRN	String		1	
KPP	RRC	String		1	
FIRST_NAME	Name	String		1	
MIDDLE_NAME	Patronymic name	String		1	
LAST_NAME	Last name	String		1	
ORG_NAME	Name of the organization	String		1	

## 4.2. Allows to obtain information from USRIE register

### 4.2.1. Receipt of USRIE (Uniform State Register of Individual Entrepreneurs) entry data

GET <endpoint>/<version>/reestr/egrip

#### *Method parameters*

Input information is not required

#### *Output information*

Output information on the IE (Individual Entrepreneur)

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
id	USRLE entry identifier	String		1	
inn	Legal body's INN	String		1	
FIRST_NAME	Name	String		1	



MIDDLE_NAME	Patronymic name	String		1	
LAST_NAME	Last name	String		1	

#### 4.3. Allows to obtain information from register of certified branches and representative offices

##### 4.3.1. Receipt of a RAPF (Register of Accredited Branches and Representative Offices of Foreign Legal Entities) register entry

GET <endpoint>/<version>/reestr/rafp

##### *Method parameters*

Input information is not required

##### *Output information*

Output information is the data on the accredited branch and representative office:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
id	USRLE entry identifier	String		1	
inn	Legal body's INN	String		1	
FIRST_NAME	Name	String		1	
MIDDLE_NAME	Patronymic name	String		1	
LAST_NAME	Last name	String		1	
KPP	RRC	String		1	

#### 4.4. Receipt of information from the register of taxes payable

##### 4.4.1. Receipt of debt information

GET <endpoint>/<version>/reestr/dues

##### *Method parameters*

Input information is not required

##### *Output information*

Output information contains an array of objects with data on each register entry:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
id	USRLE entry identifier	String		1	
inn	Legal body's INN	String		1	
BACKLOG_SUM	Amount of debt	String		1	

#### 4.5. Allows to obtain information from FIAS register

##### 4.5.1. Receipt of a FIAS object by address object identifier

GET <endpoint>/<version>/reestr/fias/addrobj/:addrobj

##### *Method parameters*

This method has the following input parameters:

Parameter	Type	Restrictions	Description
addrobj	String	GUID	Global unique identifier of an address object

##### *Example use*

##### **Request:**

GET http://<endpoint>/<version>/reestr/fias/addrobj/353b7aed-0f1b-4f44-8ce3-245083e17526

##### **Reply:**

```
{
  "CENTSTATUS": "0",
  "REGIONCODE": "01",
  "PREVID": "06e39bc5-f190-41c4-af0f-b87a990f6bf7",
  "PLAINCODE": "010030000240001",
  "POSTALCODE": "385336",
  "CURRSTATUS": "2",
  "IFNSUL": "0101",
  "EXTRCODE": "0000",
  "STREETCODE": "0001",
  "ACTSTATUS": "0",
  "_id": "353b7aed-0f1b-4f44-8ce3-245083e17526",
  "OKATO": "79218820003",
  "UPDATEDATE": "2015-02-03",
  "ENDDATE": "2015-02-02",
}
```

```

"IFNSFL": "0101",
"AOLEVEL": "7",
"SHORTNAME": "str.",
"STARTDATE": "2014-01-04",
"NEXTID": "52ae9761-4b20-4334-9163-949a39485914",
"FORMALNAME": "Shirokaya",
"CITYCODE": "000",
"AREACODE": "003",
"OFFNAME": "Shirokaya",
"OKTMO": "79618420111",
"LIVESTATUS": "0",
"PARENTGUID": "03614edb-f287-4b59-a3b3-056e160d1035",
"OPERSTATUS": "21",
"SEXTCODE": "000",
"CTARCODE": "000",
"PLANCODE": "0000",
"AUTOCODE": "0",
"PLACECODE": "024",
"AOID": "9890d854-0056-49cf-a1f2-4410e464ba9e"
}

```

#### 4.5.2. Receipt of a FIAS object by address object identifier

GET <endpoint>/<version>/reestr/fias/house/:houseobj

##### Method parameters

This method has the following input parameters:

Parameter	Type	Restrictions	Description
houseobj	String	GUID	Global unique house identifier

##### Example use

###### Request:

```
GET http://<endpoint>/<version>/reestr/fias/house/ba1c2f28-a455-47e2-95e5-000003a0023d
```

###### Reply:

```

{
"OKATO": "61226861001",
"STATSTATUS": "0",
"ENDDATE": "2014-01-04",
"IFNSFL": "6225",
"STARTDATE": "1900-01-01",
"DIVTYPE": "0",
"HOUSEID": "ba1c2f28-a455-47e2-95e5-000003a0023d",
"STRSTATUS": "0",
"UPDATEDATE": "2012-03-15",
"OKTMO": "61626424",
"TERRIFNSUL": "6212",
"COUNTER": "2",
"POSTALCODE": "391483",

```

```

    "TERRIFNSFL": "6212",

    "HOUSENUM": "2",
    "AOGUID": "fce962f2-dff8-4eea-8413-5c94e0e69dec",
    "_id": "ba1c2f28-a455-47e2-95e5-000003a0023d",
    "ESTSTATUS": "2",
    "IFNSUL": "6225"
}

```

#### 4.5.3. Receipt of a text address by FIAS identifiers

POST <endpoint>/<version>/reestr/fias/resolve

##### *Method parameters*

This method has the following input parameters:

Parameter	Type	Restrictions	Description
aoguid	String	GUID	1
houseguid	String	GUID	1
room	String	1-50 symbols	0

##### *Example use*

##### **Request:**

```

POST http://<endpoint>/<version>/reestr/fias/resolve
{
  "aoguid": "353b7aed-0f1b-4f44-8ce3-245083e17526",
  "houseguid": "ba1c2f28-a455-47e2-95e5-000003a0023d",
  "room": "10"
}

```

##### **Reply:**

```

{
  "address": "The Republic of Adygea, Krasnogvardeysky District, Chumakov
    village, Shirokaya str., build. 2, 10",
  "code": 0
}

```

#### 4.6. Receipt of information from the register of Manufacturing Licenses

##### 4.6.1. Receipt of information on Manufacturing Licenses

GET <endpoint>/<version>/reestr/prod\_licenses

### Method parameters

Input information is not required

### Output information

Output information contains an array of objects with data on each register entry:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
id	Identifier of a register entry	String		1	
inn	Legal body's INN	String		1	
ORG_NAME	Name of the organization which was given the license	String		1	
L_NUM	License number	String		1	
START_DATE	License validity start date	String		1	
END_DATE	Authorization expiration date	String		1	
L_STATUS	License status	String		1	
ADDRESS	License validity address	ADDRESS		1	
WORK_LIST	List of works/services as per license	Array of objects	String	1	

#### 4.7. Receipt of information from the register of licenses for pharmaceutical activities

##### 4.7.1. Receipt of information on licenses for pharmaceutical activities

GET <endpoint>/<version>/reestr/pharm\_licenses

### Method parameters

Input information is not required

### Output information

Output information contains an array of objects with data on each register entry:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
id	Identifier of a register entry	String		1	
inn	Legal body's INN	String		1	
ORG_NAME	Name of the organization which was given the license	String		1	
L_NUM	License number	String		1	
START_DATE	License validity start date	String		1	
END_DATE	Authorization expiration date	String		1	
L_STATUS	License status	String		1	
ADDRESS	License validity address	ADDRESS		1	
WORK_LIST	List of works/services as per license	Array of objects	String	1	

## 4.8. Receipt of information on places of business

### 4.8.1. Receipt of information on places of business

GET <endpoint>/<version>/reestr/branches

*Method parameters*

Input information is not required

*Output information*

Output information contains an array of objects with data on each register entry:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
branch_id	Place of business identifier	String		1	
address	Place of business address	ADDRESS		1	

4.8.2. Receipt of information on a specific place of business

GET <endpoint>/<version>/reestr/branches/{branch\_id}

*Method parameters*

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
branch_id	Place of business identifier	String		1	

*Output information*

Output information contains an entry:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
branch_id	Place of business identifier	String		1	



address	Place of business address	ADDRESS		1	
---------	---------------------------	---------	--	---	--

#### 4.9. Receipt of information on consignment warehouses

##### 4.9.1. Receipt of information on consignment warehouses

GET <endpoint>/<version>/reestr/warehouses

###### *Method parameters*

Input information is not required

###### *Output information*

Output information contains an array of objects with data on each register entry:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
warehouse_id	Identifier of a consignment warehouse	String		1	
address	Address of a consignment warehouse	ADDRESS		1	

##### 4.9.2. Receipt of information on a specific consignment warehouse

GET <endpoint>/<version>/reestr/warehouses/{warehouse\_id}

###### *Method parameters*

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
warehouse_id	Identifier of a consignment warehouse	String		1	

###### *Output information*

Output information contains an array of objects with data on each register entry:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
warehouse_id	Place of business identifier	String		1	
address	Address of a consignment warehouse	ADDRESS		1	

#### 4.10. Allows to obtain information from ESKLP register

##### 4.10.1. Filtering by register

POST <endpoint>/<version>/reestr/esklp/filter

##### *Method parameters*

This method has the following input parameters

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
REG_DATE	State registration date	String	Datetime	0	
REG_END_DATE	Marketing authorization expiration date	String	Datetime	0	
reg_id	Marketing authorization number	String		0	Partial access with no case sensitivity
REG HOLDER	Name of the marketing authorization holder	String		0	Partial access with no case sensitivity

PROD_SELL_NAME	Trade name of the medicinal product	String		0	Partial access with no case sensitivity
PROD_NAME	International non-proprietary name or generic (chemical) name	String		0	Partial access with no case sensitivity
REG_HOLDER_CODE	Code of a taxpayer and marketing authorization holder for RF residents	String		0	Partial access with no case sensitivity
REG_STATUS	Marketing authorization validity	String		0	Partial access with no case sensitivity
start_from	Index of the first document in delivery	Number		1	
count	The number of document in delivery	Number		1	

*Output information*

Output information contains an array of objects with data on each register entry:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
entries	Register entries	Array of objects	InfoEsklp	1	

total	Overall number of documents upon the request	Number		1	
-------	--	--------	--	---	--

Structure of InfoEsklp data:

Attribute name in JSON format	Attribute description	JSON data type	Restrictions	Cardinality	Notes
id	Unique identifier				
reg_id	Marketing authorization number	String			
PROD_NAME	International non-proprietary name or generic (chemical) name (string representation)	String			
REG_HOLDER_CODE	MA holder code	String			
PROD_PACK_1_ID	Primary packaging	String			
PROD_PACK_1_ED	Unit of mass/volume in primary packaging	String			
PROD_PACK_1_ED_NAME	Unit of mass/volume in primary packaging (string representation)	String			
PACK_2_3_NAME	Name of the site packing medicinal products into secondary/tertiary	String			

	ary packaging				
PACK_2_3_CODE	Taxpayer code of the site packing medicinal products into secondary/tertiary packaging for RF residents	String			
PACK_2_3_CODE_F	Taxpayer code of the site packing medicinal products into secondary/tertiary packaging in the registration country	String			
COUNTRY_PACK_2_3	Registration country of the site packing medicinal products into secondary/tertiary packaging	String			
QA_CODE	Taxpayer code of the site conducting the release quality stage for RF residents	String			
QA_CODE_F	Taxpayer code of the site conducting the release quality stage in the registration country	String			
PROD_PACK_2_ID	Secondary (consumer) packaging	String			
PROD_PACK_2_NAME	Secondary (consumer) packaging (string representation)	String			

QA_COUNTRY	Registration country of the manufacturer conducting the release quality control stage	String			
REG_COUNTRY	Registration country of the marketing authorization holder	String			
QA_NAME	Name of the manufacturer conducting release quality control	String			
PROD_PACK_1_SIZE	Mass/volume in primary packaging	String			
PROD_D	Number of medicinal product dosage units	String			
QA_ADDRESS_NAME	Address of the site conducting the release quality control stage (string)	String			
ADDRESS_FIAS	Address of the site packing medicinal products into secondary/tertiary packaging (according to FIAS for RF residents)	String			
QA_ADDRESS_FIAS	Address of the site conducting the release quality control stage (according to FIAS for RF residents)	String			
ADDRESS	Address of the site packing/filling medicines into secondary (tertiary) packaging	String			

REG_HOLDER_CODE_F	Taxpayer code of the MA holder in the registration country, or its analog	String			
TN_VED	TN VED code	String			
PROD_D_NAME	Number of medicinal product dosage units (string representation)	String			
PROD_FORM_NAME	Dosage form (string representation)	String			
PROD_ID	International non-proprietary name or generic (chemical) name	String			
PROD_PACK_1	Quantity of the dosage form in primary packaging	String			
PROD_SELL_NAME	Trade name of the medicinal product	String			
MAX_GNVLP	Registered price limit (for VEMs) (RUB)	String			
PROD_PACK_1_2	Quantity of primary packaging	String			

	in consumer packaging				
REG_DATE	State registration date	String			
REG_END_DATE	Marketing authorization expiration date	String			
REG_HOLDER	Name of the marketing authorization holder	String			
GNVLP	Attribute of presence on the VEM list	String			

## 5. Examples of authorization on test participants.

### 5.1. Test data of Participant No.1

#### Organization data

```
{
  "inn": "7720672100",
  "first_name": "Ivan",
  "safe_warehouses": [
    "00000000000517"
  ],
  "middle_name": "Alekseevich",
  "op_date": {
    "$date": "2017-06-02T17:02:34Z"
  },
  "last_name": "Aptechny1",
  "branches": [
    "00000000000374"
  ],
  "id": "6be50ba4-c20c-4b90-90a4-c6edbb97fe06"
}
```

#### Branches

```
{
  "_id" : LUUID("dc7c0513-9608-4708-8c1c-7889544e88e5"),
```



```
"system_subj_id" : "6be50ba4-c20c-4b90-90a4-c6edbb97fe06",
"branch_id" : "00000000000374",
"branch_address" : {
  "aoguid" : "00000000-0000-0000-0000-000000000000",
  "houseguid" : "5a46870d-7b9b-4f1c-92fd-489ef50c7811"
},
"op_date" : ISODate("2017-06-08T17:17:54.000Z")
}
```

### Consignment warehouses

```
{
  "_id" : LUUID("7d4c9e56-1f64-4f55-9190-33910bc9a24b"),
  "warehouse_org_inn" : "7732121736",
  "warehouse_id" : "00000000000517",
  "system_subj_id" : "6be50ba4-c20c-4b90-90a4-c6edbb97fe06",
  "warehouse_address" : {
    "houseguid" : "5704f7df-be84-41e0-8e89-086e43ecb641",
    "aoguid" : "00000000-0000-0000-0000-000000000000"
  },
  "op_date" : ISODate("2017-08-03T09:36:49.000Z")
}
```

### Medicinal products

```
{
  "_id" : LUUID("dd56af52-7514-4818-bb5e-fcc812f06585"),
  "gtin" : "04607143560390",
  "regDate" : ISODate("2010-06-10T00:00:00.000Z"),
  "regNum" : "ЛС-000613",
  "system_subj_id" : "6be50ba4-c20c-4b90-90a4-c6edbb97fe06",
  "op_date" : ISODate("2017-06-08T14:33:23.000Z")
}
```

## 5.2. Example of authorization for test participant No.1

/auth

```
-----
{
  "client_secret": "c4bf1684-eb4e-4119-bed7-b28fc3beb68b",
  "client_id": "ef77a1f8-e374-451d-9da9-7c3519d0d143",
  "user_id": "test_non_resident",
  "auth_type": "PASSWORD"
}
```

client\_secret and client\_id were received as a result of registering a test AS for this system\_subj\_id.

/token

```
-----
{
  "code" : "fdb7c6ba-9edf-4ee1-b09d-391d9d0430ef",
  "password" : "password"
}
```

```
}
```

code - taken from the reply to /auth

### 5.3. Test data of Participant No.2

#### Organization data

```
{
  "inn": "7826043900",
  "last_name": "Ivanov",
  "safe_warehouses": [
    "00000000000499"
  ],
  "middle_name": "Petrovich",
  "op_date": {
    "$date": "2017-09-13T17:52:57Z"
  },
  "first_name":
  "Peter", "branches": [
    "00000000000453"
  ],
  "id": "13baa6c6-e26d-4013-a01f-9908fa7df7aa"
}
```

#### Branches

```
{
  "_id" : LUUID("3f29a8e6-d7b5-4480-90e9-f2f9c6c2897c"),
  "branch_address" : {
    "aoguid" : "00000000-0000-0000-0000-000000000000",
    "houseguid" : "5a46870d-7b9b-4f1c-92fd-489ef50c7811"
  },
  "system_subj_id" : "13baa6c6-e26d-4013-a01f-9908fa7df7aa",
  "branch_id" : "00000000000453",
  "op_date" : ISODate("2017-07-26T13:13:46.000Z")
}
```

#### Consignment warehouses

```
{
  "_id" : LUUID("61770e3f-bd80-4cd6-ad66-0067e40574e8"),
  "warehouse_org_inn" : "7705772224",
  "warehouse_id" : "00000000000499",
  "system_subj_id" : "13baa6c6-e26d-4013-a01f-9908fa7df7aa",
  "warehouse_address" : {
    "houseguid" : "ed93eae1-1d65-405c-8255-38417dd6adea",
    "aoguid" : "00000000-0000-0000-0000-000000000000"
  },
  "op_date" : ISODate("2017-08-02T19:46:28.000Z")
}
```

## Medicinal products

```
{
  "_id" : LUUID("23ac8aad-b8c2-4c0b-bc7f-ecefc5211a80"),
  "system_subj_id" : "13baa6c6-e26d-4013-a01f-9908fa7df7aa",
  "regNum" : "ЛП-003403",
  "op_date" : ISODate("2017-07-24T10:10:15.000Z"),
  "gtin" : "04607028394287",
  "regDate" : ISODate("2015-12-31T00:00:00.000Z")
}
```

### 5.4. Example of authorization for test participant No.2

/auth

```
-----
{
  "client_secret": "99f360f3-fc03-40bb-adeb-1ed436dad3ca",
  "client_id": "eee32b7b-ba93-4dec-9121-56e77315aed8",
  "user_id": "test_non_resident2",
  "auth_type": "PASSWORD"
}
```

client\_secret and client\_id were received as a result of registering a test AS for this system\_subj\_id.

/token

```
-----
{
  "code" : "e41cf8b9-db20-4d80-bda0-42febac7fc4d",
  "password" : "password"
}
```

code - taken from the reply to /auth